

11-1700

A

Please type a plus sign (+) inside this box → ☐

PTO/SB/05 (08-00) (modified)

Approved for use through 9/30/2001, OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

UTILITY PATENT APPLICATION TRANSMITTAL <i>(only for new nonprovisional applications under 37 CFR 1.53(b))</i>	Attorney Docket Number	5481 US
	First Named Inventor	Guttman
	Title	Client Side, Web-Based Calculator
	Express Mail Label No.	EL566199665US

APPLICATION ELEMENTS	ACCOMPANYING APPLICATION PARTS
1. <input checked="" type="checkbox"/> Fee Transmittal Form (in duplicate) 2. <input type="checkbox"/> Applicant claims small entity status. See 37 CFR 1.27 3. <input checked="" type="checkbox"/> Specification <i>Total Pages</i> 49 <i>(preferred arrangement set forth below)</i> <input type="checkbox"/> Descriptive Title of the Invention <input type="checkbox"/> Cross Reference(s) to Related Case(s) <input type="checkbox"/> Statement Regarding Fed sponsored R & D <input type="checkbox"/> Background of the Invention <input type="checkbox"/> Brief Summary of the Invention <input type="checkbox"/> Brief Description of the Drawing(s) <input type="checkbox"/> Detailed Description <input type="checkbox"/> Claim or Claims <input type="checkbox"/> Abstract of the Disclosure 4. <input checked="" type="checkbox"/> Drawing(s) (35 U.S.C. <i>Total Sheets</i> 14 113) 5. Oath or Declaration a. <input checked="" type="checkbox"/> New Declaration <i>Total Pages</i> 3- <input type="checkbox"/> Executed (original or copy) b. <input type="checkbox"/> Copy from a prior application (37 CFR 1.63(d)) <i>(for continuation/divisional with Box 17 completed)</i> i. <input type="checkbox"/> DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b). 6. <input type="checkbox"/> Application Data Sheet. See 37 CFR 1.76	7. <input type="checkbox"/> Assignment Papers (cover sheet & document(s)) 8. <input type="checkbox"/> Certified Copy of Priority Document(s) <i>(if foreign priority is claimed)</i> 9. <input type="checkbox"/> Information Disclosure Statement & PTO-1449 <input type="checkbox"/> Copies of IDS Citation(s) 10. <input type="checkbox"/> Preliminary Amendment 11. <input checked="" type="checkbox"/> Return Postcard 12. <input type="checkbox"/> 13. <input type="checkbox"/> 14. <input type="checkbox"/> 15. <input type="checkbox"/> 16. <input type="checkbox"/>
ADDRESS TO: Box Patent Application Commissioner for Patents Washington, D.C. 20231	

17. If a **CONTINUING APPLICATION**, check appropriate box and supply the requisite information below and in a preliminary amendment or in an Application Data Sheet under 37 CFR 1.76:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: ____/____

Prior application information: Examiner: _____ Group/Art Unit: _____

For CONTINUATION OR DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 5b, is considered a part of the disclosure of the accompanying continuing or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

18. CORRESPONDENCE ADDRESS

<input type="checkbox"/> Customer Number or Bar Code Label <i>(Insert Customer No. or Attach bar code label here)</i>		Or <input type="checkbox"/> Correspondence address below	
NAME	Laura A. Majerus, Fenwick & West LLP		
ADDRESS	Two Palo Alto Square		
CITY	Palo Alto	STATE	CA
COUNTRY	U.S.A.	TELEPHONE	(650) 858-7152
		FAX	(650) 494-1417
Name (Print/Type)	Laura A. Majerus	Registration No. (Attorney/Agent)	33,417
Signature	<i>Laura Majeri</i>	Date	November 15, 2000

0002/PTO(modified) U.S. Department of Commerce
Rev. 10/2000 Patent and Trademark Office

FEE TRANSMITTAL

TOTAL AMOUNT OF PAYMENT

Subtotal (1) + Subtotal (2) + Subtotal (3) = **(\$ 1076.00)**

Complete if Known

Application Number	To Be Assigned
Filing Date	Herewith
First Named Inventor	Guttman
Group Art Unit	To Be Assigned
Examiner Name	To Be Assigned
Attorney Docket Number	5481 US

METHOD OF PAYMENT

1. The Commissioner is hereby authorized to:

- ☐ Charge the indicated fees to the below mentioned deposit account.
- ☒ Charge any additional fee required under 37 CFR 1.16 - 1.21 or credit any over payments to the below mentioned deposit account. †
- ☐ Applicant claims small entity status See 37 CFR 1.27

Deposit Account Number: 19-2555
Deposit Account Name: FENWICK & WEST LLP

A Duplicate Copy of this authorization is attached

2. ☒ Payment Enclosed:

☒ Check ☐ Credit Card ☐ Other

FEE CALCULATION (fees effective 10/01/2000)

1. FILING FEE

Large Entity Fee Code/Fee	Small Entity Fee Code/Fee	Fee Description	Fee Due
101/\$710	201/\$355	Utility Filing	710
106/\$320	206/\$160	Design Filing	
108/\$710	208/\$355	Reissue	
114/\$150	214/\$75	Provisional Filing	
SUBTOTAL (1)			(\$ 710.00)

2. CLAIMS

Large Entity Fee Code/Fee	Small Entity Fee Code/Fee	Fee Description
103/\$18	203/\$9	Claims in excess of 20
102/\$80	202/\$40	Independent claims in excess of 3
104/\$270	204/\$135	Multiple dependent claim
109/\$80	209/\$40	Reissue independent claims over original patent
110/\$18	210/\$9	Reissue claims in excess of 20 and over original patent

3. ADDITIONAL FEES

Large Entity Fee Code/Fee	Small Entity Fee Code/Fee	Fee Description	Fee Due
105/\$130	205/\$65	Surcharge - late filing fee or oath	
127/\$50	227/\$25	Surcharge-late provisional filing fee or cover sheet	
147/\$2,520	147/\$2,520	For filing a request for reexamination	
115/\$110	215/\$55	Extension for response within first month†	
116/\$390	216/\$195	Extension for response within second month†	
117/\$890	217/\$445	Extension for response within third month†	
118/\$1,390	218/\$695	Extension for response within fourth month†	
128/\$1,890	228/\$945	Extension for response within fifth month†	
119/\$310	219/\$155	Notice of Appeal	
141/\$1,240	241/\$620	Petition to revive unintentionally abandoned application	
142/\$1,240	242/\$620	Utility Issue Fee (Or Reissue)	
143/\$440	243/\$220	Design Issue Fee	
122/\$130	122/\$130	Petitions to the Commissioner	
126/\$180	126/\$180	Submission of Information Disclosure Statement	
179/\$710	279/\$355	Request for Continued Examination (RCE)	
581/\$40	581/\$40	Recording each patent assignment per property (times number of properties)	
146/\$710	246/\$355	Filing a submission after final rejection (37 CFR 1.129(a))	
149/\$710	249/\$355	For each additional invention to be examined (37 CFR 1.129(b))	
Other fee (specify):			
Other fee (specify):			
SUBTOTAL (3)			(\$ 0.00)

(Col. 1)		(Col. 2)		(Col. 3)		Fee		Fee Due	
For	No. of Existing Claims	minus*	Highest No Previously Paid For	=	Extra**	x		=	
TOTAL	27	minus*	20 or 0	=	7	x	18	=	126
INDEP	6	minus*	3 or 0	=	3	x	80	=	240
[] First presentation of multiple dependent claim									0

* Subtract the greater number of Col. 2

** If the difference between Col. 1 and Col. 2 is less than zero, then enter "0" in Col. 3

SUBTOTAL (2) (\$ 366)

SUBMITTED BY		Complete (if applicable)	
Typed or Printed Name	Laura A. Majerus	Reg. Number	33,417
Signature	<i>Laura Majerus</i>	Date	November 15, 2000

**APPLICATION FOR
UNITED STATES PATENT
IN THE NAME OF**

**Steve Guttman
Joe Ternasky**

for

CLIENT SIDE, WEB-BASED CALCULATOR

DOCKET NO. 22233.05481

Please direct communications to:

**Fenwick & West LLP
Two Palo Alto Square
Palo Alto, CA 94306
(650) 494-0600**

Express Mail Number EL566199665US

005111 "E3347/60

CLIENT SIDE, WEB-BASED CALCULATOR

Inventors: Steve Guttman and Joe Ternasky

5 This application claims priority under 35 U.S.C. §119(e) to U.S. Provisional
Serial Number 60/241,083, of Guttman and Ternasky, filed October 16, 2000, which is
herein incorporated by reference.

10 This application is related to U.S. Application Serial Number _____, of
Guttman and Ternasky, filed November 15, 2000, entitled "Client Side, Web-Based
Spreadsheet," which is herein incorporated by reference.

Background of the Invention

The present invention relates generally to dynamic hypertext markup language
(HTML) web pages and, more specifically, to a method for creating and viewing a
calculator web page that has selectively viewable components.

15 Currently, web pages are defined using Hyper-Text Markup Language (HTML).
Most web pages are passive. Specifically, a user just reads web pages and occasionally
fills in a field of a form, submits the form to a server, and waits for a reply. In general,
most users are not sophisticated consumers. Therefore, users want web-based programs
to be fool-proof and easy to use. That is, web users want web-based programs to be
20 designed so that the user cannot somehow damage the program or data by entering
incorrect data or by clicking on a wrong area of the displayed web page.

Summary of the Invention

Active applications written in Dynamic HTML combine the look of a traditional desktop application with the facilities of the World Wide Web (WWW). When the user is working with a spreadsheet written in Dynamic HTML, the user can create formulas, add and delete columns, and format cells. At the same time, the user has a disadvantage of mistakenly changing content of any particular cell because each cell in a spreadsheet is editable. Thus, there is a need for a method and system that will allow working with a web-based spreadsheet so that the user is unable to change the content of any cell by mistake.

A described embodiment of the present invention allows a designer/creator to create and view "calculator" web pages. A calculator web page is a special version of a spreadsheet web page. When a user is working with a spreadsheet, it is unfortunately easy for him to mistakenly change the content of a particular cell because each cell in a spreadsheet is editable. The described embodiment of the present invention allows the user to view a spreadsheet as a calculator so that the user can work with a fill-in-the-blanks, standalone application. The user can just enter his data and view the calculator results. The user cannot change the content of any cell by mistakenly clicking on that cell. By default, the calculator uses a described method to determine which spreadsheet cells are initially editable in the calculator.

Another advantage of the described embodiment of the present invention is that in order for the user to work with a calculator, no other special software is involved, except for a web browser. The present invention serves as a design tool for the calculator designer because it produces interactive HTML pages without requiring the designer to

be familiar with the web programming. In accordance with the purpose of the present invention, as embodied and broadly described herein, the invention is a computer-implemented method for viewing a web-based spreadsheet in a calculator mode. The method comprises sending a request, by a user, to view the calculator web page;

- 5 displaying the requested calculator web page, wherein only certain cells in the calculator are editable; and allowing the user to change contents in editable cells, but not allowing the user to change contents in non-editable cells.

In further accordance with the purpose of the invention, as embodied and broadly described herein, the invention is a computer-implemented method performed by a server
10 data processing system. The method comprises receiving a request, from a client, for a calculator web page; reviewing parameters received with the request for the calculator web page, said parameters include a calculator mode and a data ID; sending the requested calculator web page, to the client, based on the parameters received, wherein said calculator web page contains embedded data specific to the requested calculator web
15 page and wherein the calculator web page displays a calculator having only certain cells that are editable.

In further accordance with the purpose of the invention, as embodied and broadly described herein, the invention is a data structure comprising a member file for keeping information about members, a spreadsheet /calculator data file for keeping spreadsheet
20 and calculator information on a server, and a Data Array file for keeping spreadsheet and calculator information on a client.

In further accordance with the purpose of the invention, as embodied and broadly described herein, the invention is a method for building a calculator, the method

comprising determining a number of rows and columns for the calculator; initializing the content of each cell in the calculator; making the cells visible once all the cells are built; and performing calculations for cells having formulas in them.

Advantages of the invention will be set forth in part in the description which follows and in part will be apparent from the description or may be learned by practice of the invention. The objects and advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims and equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate several embodiments of the invention and, together with the description, serve to explain the principles of the invention.

Fig. 1 pictorially illustrates a web-based spreadsheet in a spreadsheet mode in accordance with a described embodiment of the present invention.

Fig. 2 illustrates a web-based calculator created using calculator preview mode in accordance with a described embodiment of the present invention.

Fig. 3 pictorially illustrates adjusting editability of cells in the calculator of Fig. 2 using calculator preview mode in accordance with a described embodiment of the present invention.

Fig. 4 pictorially illustrates the spreadsheet of Fig. 1 during use in a calculator mode in accordance with a described embodiment of the present invention.

Fig. 5 is a block diagram illustrating an overall architecture in accordance with a described embodiment of the present invention.

Fig. 6A is a flow chart showing the process of returning an HTML web page to the user, performed by a server, in accordance with a described embodiment of the present invention.

Fig. 6B is a flow chart showing the steps performed by a browser in accordance with the present invention.

Fig. 7 is a flow chart illustrating the process of building a calculator in accordance with a described embodiment of the present invention.

Fig. 8 is a block diagram showing an example record in a member file in accordance with a described embodiment of the present invention.

Fig. 9 is a block diagram showing an example record in a spreadsheet /calculator data file in accordance with a described embodiment of the present invention.

Fig. 10 pictorially illustrates a process of allowing a user to embed a calculator in any web page in accordance with a described embodiment of the present invention.

Fig. 11 is a flow chart showing a log in process in accordance with a described embodiment of the present invention.

Fig. 12 is a flow chart showing a process of saving data to a server in accordance with a described embodiment of the present invention.

Fig. 13 pictorially illustrates a “My Files” page containing saved files in accordance with a described embodiment of the present invention.

Detailed Description of Illustrated Embodiments

5 Reference will now be made in detail to several embodiments of the present invention, examples of which are illustrated in the accompanying drawings. Wherever practicable, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

10 A described embodiment of the present invention allows a user to view a web-based spreadsheet as a calculator, wherein only certain cells of the calculator are editable. The designer/creator of the web-based spreadsheet allows the user to alter the default as to which cells are editable when the calculator is displayed in a calculator preview mode. The described embodiment further allows the user to change the cells’ content in the calculator mode based on the adjustments made in the calculator preview.

15 Fig. 1 illustrates a web-based spreadsheet in a spreadsheet mode. The spreadsheet 100 contains columns and rows where all the spreadsheet calculations are performed. Each row has a number and each column has a letter. A cell is the intersection of a row and a column and is referenced with a Column letter Row number notation, such as A1 or C3. The example spreadsheet 100 has at least 20 rows (numbered 1 through 20) and 5
20 columns (labeled A through E). The user can reset the number of rows and columns in the spreadsheet with the “Set Size” command. In addition, the user can add or delete columns with the “Insert/Delete Rows or Columns” commands. A cell can contain labels, numbers, text (strings), dates/times, or formulas. Other embodiments may have

additional types of data in their cells. A string is a text entry, such as, “Make Thanksgiving dinner,” as displayed in cell A1 (104). A number is an integer or decimal value, which can be formatted as currency, integers (whole numbers), decimal numbers, or percentages. Users can define arbitrary formats.

- 5 The user can change information for any specific cell in the spreadsheet. To do so, the user has to click on that cell to highlight it. The command line 102 is the area of the spreadsheet where a user enters and edits cell values and formulas. To change information for a specific cell, a user clicks on that cell to highlight it. Then, the user can type the value, text or formula for that cell into the command line field. In the illustrated
- 10 spreadsheet 100 the user entered the number “12” into the command line for cell B3 (106) and “2” for cell B5 (108). Also, the user chose a “traditional” dinner in cell 110 over “economy” or “alternate” dinner. If the user wants to change the type of meal he is planning (for example, the user wants to type “Corn Beef” instead of “Roast Turkey”), the user just needs to click on that cell to highlight it and enter “Corn Beef” into the
- 15 command line. In the example spreadsheet, cell 111 has the following associated formula, as displayed in the command line 102: **=IF(\$B\$7="Traditional",1.25*(B3-B5),IF(\$B\$7="Alternate",(B3-B5),IF(\$B\$7="Economy",(B3-B5)*2,"")))**. Thus, if cell B7 contains “Traditional,” a value of 12.5 is displayed in cell B13 (111), (as shown). This value is derived from the formula **1.25*(B3-B5)**, wherein **B3=12** and **B5=2**.
- 20 Different values of cell B7 cause other values to be displayed in cell B13 (111). Cells 112 also have formulas associated with them. Working with a spreadsheet allows a user to customize the spreadsheet for whatever purpose the user desires. When a user is

working with a spreadsheet, however, it is easy for him to mistakenly change content or formulas of a particular cell because each cell in a spreadsheet is editable.

Fig. 2 illustrates a web-based calculator created using calculator preview mode in accordance with a described embodiment of the present invention. Any spreadsheet can be viewed in either spreadsheet mode, calculator mode, or calculator preview mode. The drop-down “Calculator Preview” command, which is accessed from the spreadsheet mode, lets a designer/creator adjust which cells may be “filled-in” (edited) when the spreadsheet is viewed as a calculator. Viewing the spreadsheet in a spreadsheet mode, as illustrated in Fig. 1, allows a user to customize the spreadsheet for any purpose the user wishes. To pre-view the spreadsheet as a calculator, the user has to select the “Calculator Preview” command from the Tools menu. As shown in Fig. 2, the Menu Bar, Format Bar and Command Line will disappear. In place of the Format Bar an “Exit Preview” button (202) will appear. The spreadsheet grid is also hidden, and all cells that may be edited by a user of the calculator (106, 108, and 110) have a blue outline (or similar indication) around them.

By default, the system will take its best guess at which cells should be editable when the spreadsheet is displayed in a calculator preview mode. Cells that depend on values in other cells (for example, a cell with an associated formula (111)) default to non-editable. Cells with values that other cells depend on (106, 108, and 110) (but which don’t have formulas in them) default to editable. Cells that do not depend on values in other cells default to non-editable.

Fig. 3 illustrates the process of allowing a designer/creator to adjust editability of cells in the spreadsheet of Fig. 1 using the calculator preview mode in accordance with a described embodiment of the present invention. As was illustrated in Fig. 2, by default, the system will take its best guess at which cells should be editable when the spreadsheet is initially displayed in calculator preview mode. Each Data Array file 502 and Spreadsheet/Calculator Data file 516 (as will be discussed with reference to Fig. 5) includes editability flags for each cell, which is assigned a "FALSE" or "TRUE" logic value. This value indicates whether a certain cell is editable or not. If the cell has a "TRUE" value, it is locked, and a user cannot edit that cell in the displayed calculator, in the calculator mode. Alternatively, if the cell has a "FALSE" value, it is unlocked, and a user of the displayed calculator can edit that cell. Thus, for example, in the displayed calculator of Fig. 4, only cells 106, 108, and 110 are unlocked and can be changed by a user in the calculator mode.

In some cases, however, a designer of a spreadsheet/calculator may want to adjust the default editability. To do so, the calculator designer/editor needs to click on a cell in calculator preview mode to toggle it from editable to non-editable and vice versa. Once the designer/creator clicks on the cell to adjust its editability, the matching object is found in the HTML page, and the executing JavaScript changes the editability flag value from "TRUE" to "FALSE" (or vice versa). For example, as shown in Fig. 2, a user viewing the calculator could only edit the cells related to the number of guests 106, the number of vegetarians 108, and the type of meal 110. Adjusting the cells' editability allows the user to edit the descriptive text or any other cell in the calculator when the calculator is displayed to the user. Once the designer/creator adjusts editability of cells, a user can

change the cells' values in a calculator mode. In particular, if the designer/creator toggles cell 304, which includes the text "Roast Turkey," in calculator preview mode the designer/creator changes its editability in the calculator preview mode so that a blue outline (or similar indication) appears around cell 304. Once in a calculator mode, the user can change the value of cell 304 by typing "Corn Beef" instead of "Roast Turkey". Similarly, toggling cell 302 in the calculator preview mode by the designer/creator allows the user to change the content of cell 302 in the calculator mode from "How many vegetarians" to "How many lacto-vegetarians." Thus, once the designer/creator has finished these modifications to editability, the user can change values of cells in the calculator mode based on the editability adjustments made in the calculator preview.

Fig. 4 illustrates an example of displaying the spreadsheet of Fig. 1 in a calculator mode in accordance with a described embodiment of the present invention. The example calculator web page 400 features a calculator displayed in calculator mode and in which only certain cells are editable. The calculator 400 allows the user to plan the Ultimate Holiday Dinner based on the number of guests and type of meal the user would like to make. The name of the calculator is displayed on the top of the page in the form of a text entry: "Make Thanksgiving dinner 1" 402. The user is prompted to enter information related to the number of total guests and the number of vegetarians among the guests and 108. The user also is prompted to choose among traditional, alternate, or economy dinner 110. A number of total guests, number of vegetarians and a type of the dinner are the only editable cells in the calculator. These cells were set to be editable either by default when the spreadsheet was made or by the designer/creator in calculator preview mode.

In the bottom half of the page, the calculator displays the assortment of food that the user needs to have in order to prepare the dinner. It includes the following text entries, which are not editable: Roast Turkey, Bread Stuffing, Soup, Side of Fruits and Vegetables, Side of Potatoes, Crab Cake, and Cranberry Juice 406. In the calculator 400, the user is having twelve guests (where two people are vegetarians). The user has chosen a traditional dinner (110). Once the user enters all the requested information, the calculator estimates the quantity of food that the user needs to have in his kitchen in order to accommodate the total number of invited guests. It should be noted that the cells displaying the quantity of food 408 are not editable by the user. The calculator was so-designed by its designer/creator. The calculator determines that for the traditional type of dinner the user needs to have 12.5 pounds of roast turkey, 12 cups of bread stuffing, 12 cups of soup, 12 cups of vegetables on a side, 6 pounds of potatoes on a side, 2 pumpkin pies, and 24 cups of hot apple cider. Each of these cells has an associated formula depending on one or more of cells 106, 108, 110. The example calculator web page also includes a function menu 410 located at the top of the page allowing the user to save the calculator, print the calculator, receive instructions, edit calculator as a spreadsheet, email the calculator to others, and rate the calculator. In order to email the calculator, the user should click on “email this page” button and enter a recipient’s address. The system will email a URL of the calculator, including the proper parameters in the URL.

20 Thus, the present invention advantageously allows working with a spreadsheet that is displayed in a calculator mode so that the user cannot change the content of any cell by mistakenly clicking on that cell. The calculator designer/creator can change the system default as to which cells are editable.

Fig. 5 is a block diagram illustrating an overall architecture of the present invention. A server system 510 includes, but is not limited to, a Member file 512, server software 514, a Spreadsheet /Calculator data file 516, a spreadsheet HTML 518, and a calculator HTML 521. It should be understood that the architecture illustrated in Fig. 5 is shown for purposes of example only and is not to be construed in a limiting sense.

The Member file 512 includes information for all members who have an account with the system. Initially, when a user logs into the system, he is asked whether he is a member who has an account with the system (the “log in” process will be discussed in detail with reference to Fig. 11). If the user is a member, he is allowed to access the system. In the alternative, if the user is not a member, he is offered to enter his identifying information, which might include a password and email address.

The Spreadsheet /Calculator data file 516 contains information related to spreadsheets and calculators. Such information may include a data ID, which refers to a file name, and information for each individual cell of a spreadsheet/calculator. In particular, the cell information related to each individual cell includes, but is not limited to, the information related to cell dependency, formatting, content, and editability. The server software 514 manages the files and communicates with the browser. The server software 514 handles opening, saving, and incorporating live data (such as stock quotes) into the files. In addition, the server software 514 returns the HTML page 518 and calculator page 521 in response to a request received from the client system 520, via a browser 504.

The client system 520 includes browser software 504 and a Data Array 502. The user, via the browser 504, sends a request to the server 510 for an HTML page. The

server software 514 reviews the parameters received with the request. If the parameters include a "spreadsheet" mode, a spreadsheet HTML page 518 is returned to the client 520. In the alternative, if the parameters include a "calculator" mode, the server 510 returns the calculator HTML page 520. If the parameters include an "embed" mode, the server returns a fully formed calculator web page. The Data Array 502 is an array of cell descriptions whose values are loadable from JavaScript in the HTML page. The Data Array 502 contains information related to each individual cell, which includes, but is not limited, to cell dependency, formatting, content, and editability. When the designer/creator saves a file, client 520 turns the definition of each cell into a string. The concatenated cell strings define the entire spreadsheet. Client 520 sends the concatenated string to the server 510. The server 510 takes the string and writes it into the Spreadsheet/Calculator Data File 516. Thus, a client-format Data Array 502 is translated into Spreadsheet/Calculator Data File 516 in the server file format.

The browser 504 is software effecting the requesting and displaying of HTML web pages. The browser software can be standalone or integrated within other software products. It should be understood that each of clients and web servers in the described embodiment preferably includes a processor and a memory. The memory includes instructions capable of being executed by the processor to perform the functions described below. The server can also include a computer readable medium for storing the instructions. The server system 510 communicates with the client system 520 via any appropriate communication mechanism, including but not limited to, a network, an intranet, the Internet, wireless communications, telecommunications, cable modems, and satellite communications.

DocId: 3460

Fig. 6A is a flow chart showing the process of returning an HTML web page, performed by the server 510, in accordance with a described embodiment of the present invention. In element 602, the server 510 receives a request for an HTML page, from the browser 504. In element 604, the server reviews the parameters received with the request (for example, the URL parameters in Common Gateway Interface (CGI) style). The parameters may include a data ID, which refers to the file name, and a mode, which preferably is one of spreadsheet mode, calculator mode, or embed mode. The server 510 finds the data in element 606 based on the data ID parameter received. If the designer/creator requested a spreadsheet, in element 608, the server embeds data using JavaScript and returns a spreadsheet HTML web page 518 to the designer/creator in element 610. If the designer/creator requested a calculator in element 612, the server 510 returns a calculator HTML web page 521 to the designer/creator in element 614. The spreadsheet HTML 518 contains script tags, spreadsheet data, and an HTML user interface. The calculator HTML 521 contains script tags, calculator data, and an HTML user interface. For example, in the case of a calculator HTML page 521, the script tag states the following, as shown in Table 1 (Table 1 shows an example of JavaScript included within a calculator web page, and is herein incorporated by reference):

href="/css/calculator_ie4_eda7fc0b.cssx", whereas a spreadsheet HTML page 518 script tag states: **href="/css/spreadsheet_ie4_630ddc74.cssx"**. Furthermore, in the calculator HTML page 521, the user interface differs from that in a spreadsheet HTML 518. There are also some differences in JavaScript code embedded in calculator HTML page 521 and spreadsheet HTML 518. In particular, cells in a calculator HTML page 521 have different parameters than cells in a spreadsheet HTML page 518. Thus, cells in a

spreadsheet are unlocked and the editability flag has a “FALSE” value, whereas most cells in a calculator are locked and the editability flag has a “TRUE” value.

If the designer/creator requested data in the Embed mode, a loaded, fully formed HTML page is returned, in element 616. This HTML page does not include HTML script tags, unlike calculator HTML 521 and spreadsheet HTML 518. The embedded HTML page dynamically returns the HTML page. This is accomplished using “JavaScript include” tags. The “JavaScript include” tags return dynamically created JavaScript reflecting the current change in the calculator or spreadsheet file. The process ends in 618 once the HTML page is returned to the designer/creator.

Fig. 6B is a flow chart showing the steps performed by a browser 520 in accordance with the present invention. Once the browser sends a request for an HTML page, as was discussed with reference to Fig. 6A, the server reviews the parameters received with the request and returns the HTML page accordingly. The parameters received may include a Data ID, which refers to the file name, and a mode, which preferably is one of spreadsheet mode, calculator mode, or embed mode. The browser 504, in turn, receives the HTML page, parses the page written in JavaScript, and executes JavaScript code, in element 626. The HTML page may include JavaScript files specific to spreadsheets and calculators, such as “Calculator JS file” and “Spreadsheet JS file” and shared data files, which could be shared by calculators and spreadsheets. Once the browser parses the page in element 626, it might send additional requests to the server 510 for data, which was referred to in the HTML page. In element 628, loading progress bar reports are generated and displayed informing the user about the status of the progress of program execution. In particular, the progress bar may indicate the following:

“Loading a Calculator” or “Loading a Spreadsheet.” In element 630, the calculator is built (as shown in Fig. 7). In element 632, the user has the option of saving the file he is currently working with if the data was modified since the last time the file was saved. The process ends in element 634. Building the calculator 630 is discussed below.

5 Fig. 7 is a flow chart illustrating the process of building a calculator in accordance with a described embodiment of the present invention. Initially, a client system 520 receives the number of rows and columns to build a particular calculator, in element 702. For instance, as illustrated in reference with Fig. 4, an example calculator has 6 columns and 24 rows. A two-dimensional array of cell elements is built. In element 704, each cell

10 is initialized and a dynamic HTML is specified for each cell. This step includes obtaining information about a particular cell from a Data Array file 502. This information includes, but is not limited to, cell dependency, formatting, content, and editability. Thus, as illustrated in Fig. 4, cell A3 which includes the text “HOW MANY TOTAL GUESTS ARE YOU HAVING?” is described by the following parameters, as shown in

15 Table 1: {ENTRY:'**HOW MANY TOTAL GUESTS ARE YOU HAVING?**',LOCKED:'FALSE',VIEWSIZE:'9PT',FORECOLOR:'NAVY',VIEWFAMILY:'VERDANA',_WRAPTEXT:'TRUE',_TEXTALIGN:'LEFT',M_ROW:3,M_COL:1,I_NR:'HOW MANY TOTAL GUESTS ARE YOU HAVING?'}. According to these parameters, cell A3 is the intersection of a third row and a first column and is referenced with a Column # Row #

20 notation A3. Cell A3 is an editable cell because the editability flag is set to “FALSE” (the calculator was so-designed by its designer/creator), the text in the cell is aligned to the left, the color of the cell is navy, and the size of the cell is 9pt.

Once dynamic HTML is specified for each cell, the dependency tree is built, in step 705. The dependency tree is part of the Data Array 502. Any calculator has at least two types of cells: the ones that depend on a particular cell and the ones that a particular cell depends on. Some cells in the Data Array 502 contain the field "i_rt" and others contain the field "i_tb". The fields are lists of other cells that a given cell depends on (i_rt) and a list of cells that depend on this cell (i_tb). "I_tb" stands for "initialize the referred to by cell" and "i_rt" stands for "initialize refers to cell." In the example calculator illustrated in Fig. 3, cell B13 (111) is a dependent cell. The following is the list of other cells that cell B13 depends on, as shown in Table 1:

10 I_RT:'[E[7][2],E[3][2],E[5][2]]',LOCKED:'FALSE',VIEWSIZE:'9PT',_WIDTHCLUE:'80',_T
EXTALIGN:'RIGHT',M_ROW:13,M_COL:2,I_NR:'12.5'}. Accordingly, cell B13, which is the intersection of row 13 and column 2, depends on cells B7 (110), B3 (106), and B5 (108), where cell B7 displays the type of dinner the user selected, cell B3 displays the number of guests the user is expecting, and cell B5 displays the number of vegetarians among the guests. Thus, if cell B7 contains "Traditional," a value of 12.5 is displayed in cell B13 (111), (as shown in Fig. 3). This value is derived from the formula, which is described in the Data Array 502: $1.25*(B3-B5)$. If B3=12 and B5=2, the value displayed in cell B13 is 12.5. If cell B7 contains "Economy," a value of 20 is displayed in cell B13 (not shown). This value is derived from the formula described in the Data Array 502

20 $(B3-B5)*2$. If B3=12 and B5=2, the value displayed in cell B13 is 20.

Once all cells are built, they become visible, in element 706. This is done with a combination of HTML that is generated on the fly using JavaScript and by JavaScript itself. In element 708, calculations are performed for each cell, when needed. For

example, if the user chooses a “traditional” dinner in cell B7, cell A13 displays a “Roast Turkey.” Formula calculations are performed for cell B13 to determine how many pounds of roast turkey are required to prepare the dinner based on the number of guests.

If the user chooses an “alternate” dinner in cell B7, cell A13 displays “Cornish Game

5 Hens.” Calculations are performed for cell B13 to determine how many pieces of hen are required to prepare the dinner. Similarly, if the user chooses “economy” dinner in cell B7, cell A13 displays “Chicken Parmesan.” Calculations are performed for cell B13 to determine how many pieces of chicken are required to accommodate twelve guests, where two of them are vegetarians.

10 Once all calculations are made, the determination is made whether a particular cell is editable (whether it is locked), in element 710. If a cell is non-editable, that cell is locked and the user is not allowed to enter data into that cell in calculator mode, in element 712. For example, according to Data Array 502 description, cells A13-A23 are locked and the user cannot change their content. In the alternative, if the cell is editable
15 (unlocked) (for example, cells B3 and B5, as described in the Data Array 502), an input box is built around that cell, in element 714. The user can type in the value in that cell because it is unlocked in calculator mode, in element 716. Thus, the user can type in “12” in cell B3 and “2” in cell B5. The process ends in 718.

Fig. 8 is a block diagram showing a Member file 512 and an example member
20 record 800. The Member file 512 keeps member information in the form of member records 800. The member record 800 identifies a particular member. In one embodiment of the present invention, the example member record may have the following fields: a member ID, a password, an email address, and a data ID, which refers

to a file name. The example record 800 features a member ID equal to ANNA, 0124 as a password, ANNA@HOTMAIL.COM, as an email address, and “Make Thanksgiving dinner 1” as a file name. When a user initially signs up with the system, all user information is forwarded to the member file 512.

5 Fig. 9 is a block diagram showing a Spreadsheet /Calculator data file 516 and an example record 900. The Spreadsheet /Calculator data file 516 keeps spreadsheet and calculator data information. The same data file is used to generate spreadsheet HTML 518 and calculator HTML 521. In particular, it contains a user name, data ID (for example, a file name) and information about each cell. The information for every cell
10 may include value, formula, formatting, editability, and borders. Each Data Array file 502 and Spreadsheet/Calculator Data file 516 includes an editability flag for each cell, which is assigned a “FALSE” or “TRUE” logic value. This value indicates whether a certain cell is editable or not. If the cell has a “TRUE” value, i.e., if it is locked, a user cannot edit that cell when viewing the calculator. Alternatively, if the cell has a
15 “FALSE” value, i.e., if it is unlocked, that cell can be edited when viewing the calculator. Fig. 9 features the example record 900 having the following fields: User Name = ANNA, Data ID= “Make Thanksgiving Dinner,” and information about each cell. In particular, cell A3 which includes the text “HOW MANY TOTAL GUESTS ARE YOU HAVING?” is described by the following parameters, as shown in Table 1: {ENTRY:'**HOW MANY**
20 **TOTAL GUESTS ARE YOU**
HAVING?',LOCKED:'FALSE',VIEWSIZE:'9PT',FORECOLOR:'NAVY',VIEWFAMILY:'VERDA
NA',_WRAPTEXT:'TRUE',_TEXTALIGN:'LEFT',M_ROW:3,M_COL:1,I_NR:'**HOW MANY**
TOTAL GUESTS ARE YOU HAVING?'}. According to these parameters, cell A3 is the

intersection of a third row and a first column and is referenced with a Column letter Row number notation A3. Cell A3 is an editable cell because the editability flag is set to “FALSE” (the calculator was so-designed by its designer/creator), the text in the cell is aligned to the left, the color of the cell is navy, and the size of the cell is 9pt.

5 Fig. 10 illustrates a process of allowing a web page user to embed a calculator in any web page. If a user has a personal web site, he can include one of his calculators with other text and graphics directly into his web site. To embed a calculator in a web page, the user has to go to the “My Files” page (see Fig. 13). Then, the user has to highlight the file he wants to embed and select the “Sharing” command from the Action

10 Panel on the right side of the screen. Performing this step will bring up a dialog box with the appropriate HTML code (which includes the parameters of data ID for the calculator and anchor text that will appear as a link on the page). The user has to copy this code (1002) to the clipboard and then paste the code into his web page’s HTML at the location he wants the calculator to appear. In the Sharing dialog, the user first needs to set his file

15 to Public. Embedding the calculator advantageously allows to view the calculator file within the user’s own web page. When a web page designer/creator embeds a calculator on a web page, he is actually adding some code to an HTML page which calls that calculator from the server 510. When this code is downloaded to someone's web browser, the browser makes a request to the server 510 to display the calculator of interest. The

20 server then returns the appropriate calculator and Dynamic HTML code to that web browser. As discussed above in reference to Fig. 6A, the HTML page does not include HTML script tags, unlike calculator HTML 521 and spreadsheet HTML 518. The embedded HTML page includes means to dynamically return the HTML content. These

means are “JavaScript include tags.” The JavaScript include tags return dynamically created JavaScript reflecting the current change in the calculator or spreadsheet file.

The embedded calculator is a fully formed calculator web page which looks identical to the calculator web page discussed in reference to Fig. 4. The example calculator web page 400 features a calculator, wherein only certain cells are editable. The calculator 400 allows the user to plan Thanksgiving dinner based on the number of guests and type of meal the user would like to make. The name of the calculator is displayed right on the top of the page in the form of a text entry: “Make Thanksgiving dinner” (402). The user is prompted to enter information related to the number of total guests and the number of vegetarians among the guests 106 and 108. Also, the user is prompted to choose among traditional, alternate, or economy dinner 110. A number of total guests, number of vegetarians and a type of the dinner are the only editable cells in the calculator. These cells were set to be editable either by default when the spreadsheet was made or by the designer/creator in calculator preview mode.

Fig. 11 is a flow chart showing a log in process. The process starts in element 1100 and determines whether the user is a private user, in element 1102. The user has the option of making any individual file a “public” file, so that he can share it with others (see Fig. 10). A private user can only view private files if he has logged into the account where the file is listed. Alternatively, a public user has the URL of the file and wants to be able to view the file outside the private user’s personal account. If the user is a private user, the process determines whether the user is a member, in element 1104. At this point, the user’s name and password are posted to the server. The server 510 reviews the parameters and verifies that the user has an account with the system, i.e., the user is a

member. If the user is not a member, the ERROR message is returned in element 1108, and the user is prompted to sign up with the system, in element 1118. In the alternative, if the user is a member, as determined from member file 512, the system determines, in element 1106, whether the user is the owner of the file he would like to work with. Every

5 Spreadsheet/Calculator data file 516 contains a member ID. If the user ID matches the member ID in the Spreadsheet/Calculator data file 516, the user is the file owner. If the user is a file owner the system allows the user to view that file, in element 1112, and the process ends in 1114. Similarly, if the user is a public user a public user who has the URL of the file and wants to be able to view the file outside the private user's personal

10 account, the system allows the public user to view that file, in element 1112.

If the member is not a file owner, the ERROR message is returned in element 1110 and the user is prompted to work with that particular file as not a file owner, in element 1116. If the user is a public user (he has the URL of the file and wants to be able to view the file outside the private user's personal account) he can view that file, in

15 element 1112.

Fig. 12 is a flow chart showing a process of saving data into "My Files." The *Save* and *Save As* commands, which are accessed from the FILE menu, allow a calculator designer/creator to save the file he is currently working with. *Save* will overwrite the last saved version of the file, where *Save As* renames the current file and saves it (under a

20 new name). The process starts in element 1202. In element 1204, the process determines whether calculator designer/creator changes editability. If the designer/creator changes editability, the editability flag is changed from "FALSE" to "TRUE" in client-side Data Array 502, in element 1206. If designer/creator changes cell's content in a spreadsheet,

in element 1208, data content is changed in the Data Array 502, in element 1210. If the creator/designer indicates that he wants to save new data, in element 1212, the server 510 receives a “post” in the form of new data in client-side format, from a browser, in element 1214. In element 1216, the server determines whether the creator/designer is a member who has an account with the system in accordance with member file 512. If the user is a member, the process determines, in element 1218, whether the user is the owner of the file he is working with. In the alternative, if the user is not a member, in element 1222 the user is prompted to register and become a member. If the member is the owner of the file he is working with, the process offers the user to save the information in place of the original file, in element 1220. In the alternative, if the user is not the owner of the file, he is offered to save the file as not a file owner, in element 1224. The process then saves information about each cell, including all attributes, such as format, borders, editability, and cell dependency as well as rows height and columns width into non-owner account in element 1226. The process ends in element 1228.

Fig. 13 illustrates “My Files” page containing saved files. The “My Files” page 1300 is a user’s personal file manager where each member accesses and manages his files. To access the files from the “My Files” page 1300, the user has to enter his name and password. The user has the option of making any individual file a “public” file, so that he can share it with others. The upper left panel 1302 of the page lets the user navigate between different major areas of the site. The middle area 1304 of the page displays user’s list of files. The menu bar 1306 directly above the list of files allows the user to create new files, import files and images, sort the list of files, and view the files according to their type.

WHAT IS CLAIMED IS:

1. A method for viewing a calculator web page in a calculator mode, comprising:
 - sending a request, by a user, to view the calculator web page;
 - displaying the requested calculator web page, wherein only certain cells in the calculator are editable; and
 - allowing the user to change contents in editable cells, but not allowing the user to change contents in non-editable cells.
2. The method of claim 1, further comprising e-mailing a link to the calculator, including:
 - allowing the user to click on “e-mail this page” button; and
 - allowing the user to enter a recipient’s address.
3. The method of claim 1, further comprising embedding the calculator in a web page, including:
 - allowing the user to indicate a calculator ID of the calculator;
 - displaying an HTML code including the calculator ID of the calculator; and
 - allowing the user to copy the HTML code and paste it into the web page’s HTML.
4. The method of claim 3, wherein the embedded HTML web page includes “JavaScript include” tags.

5. The method of claim 1, wherein the calculator is displayed on a client system.
6. The method of claim 5, wherein a JavaScript Dynamic HTML is downloaded to the client system, the JavaScript including functionality, format and content of the calculator web page.
7. A computer-implemented method performed by a server data processing system, comprising:
- receiving a request, from a client system, for a calculator web page;
 - reviewing parameters received with the request for the calculator web page, said parameters include a calculator mode and a data ID;
 - sending the requested calculator web page, to the client system, based on the parameters received, wherein said calculator web page contains embedded data specific to the requested calculator web page and capable of causing display of a calculator having only certain cells that are editable.
8. The method of claim 7, wherein before sending the calculator web page, JavaScript data is embedded into the calculator web page.
9. The method of claim 7, wherein the data ID identifies a file name.
10. A method for viewing a spreadsheet using a calculator preview mode, comprising:

09446314300

sending a request, from a designer/creator, to view the spreadsheet in a spreadsheet mode;

displaying the spreadsheet in the spreadsheet mode;

requesting, by the designer/creator, to view the spreadsheet using the calculator preview mode; and

displaying the spreadsheet in the calculator preview mode, wherein certain portions of the displayed spreadsheet have been selected as non-editable.

11. The method of claim 10, wherein requesting by the designer/creator, to view the spreadsheet using the calculator preview mode includes allowing the designer/creator to select the Calculator Preview command from the Tools menu.

12. A method for allowing a user to change contents of cells in a calculator in a calculator mode, the method comprising:

displaying a spreadsheet using a calculator preview mode;

allowing a designer/creator to adjust the cells' editability in the calculator preview mode;

allowing the user to change the contents of the cells in the calculator in the calculator mode based on the adjustments made in the calculator preview mode.

13. The method of claim 12, wherein the cells depending on values in other cells default to being non-editable.

14. The method of claim 12, wherein the cells with values that other cells depend on default to being editable.
15. The method of claim 12, wherein the cells containing text only default to being non-editable.
16. The method of claim 12, wherein adjusting editability includes allowing the designer/creator to toggle the editability of a cell when the designer/creator clicks on that cell.
17. The method of claim 12, wherein editable cells have a blue outline around them in the calculator preview mode.
18. The method of claim 12, wherein adjusting editability includes changing a flag value in each Spreadsheet/Calculator Data file and Data Array file.
19. The method of claim 18, wherein the flag in the Spreadsheet/Calculator Data file has a “FALSE” value when the cell is editable.
20. The method of claim 18, wherein the flag in the Spreadsheet/Calculator data file has a “TRUE” value when the cell is non-editable.

21. A data structure for allowing a user to view a calculator in a calculator mode, the data structure comprising:

a member file for keeping information about members;

a Spreadsheet /Calculator Data file for keeping a spreadsheet and calculator data information by a server; and

a Data Array file for keeping the spreadsheet and the calculator data information by a client.

22. The data structure of claim 21, wherein the member file includes a member ID, password, and data ID.

23. The data structure of claim 21, wherein the spreadsheet /calculator data file includes a user name and information about each cell.

24. The data structure of claim 23, wherein the information about each cell includes at least one of value, formula, formatting, borders, content, dependency, and editability.

25. A method for building a calculator, the method comprising:

determining a number of rows and columns for the calculator in accordance with data sent from a server;

initializing the content of each cell in the calculator in accordance with data sent from a server;

making the cells visible once all the cells are built; and

performing a formula calculation for cells having formulas in them.

26. The method of claim 25, further comprising the steps of:

building an input box for editable cells; and

allowing a user to enter contents in the editable cells, but not allowing the user to enter contents in non-editable cells.

27. The method of claim 25, wherein the step of initializing includes obtaining information about each particular cell from a Data Array file.

CLIENT SIDE, WEB-BASED CALCULATOR

ABSTRACT OF THE DISCLOSURE

A method and data structure that allows a user to view a spreadsheet in a “calculator” mode, wherein certain cells are non-editable. The calculator designer/creator can alter the default as to which cells are editable when the calculator is displayed in the calculator preview mode. Once the designer/creator made all modifications in the calculator preview mode, the user can change the cell values in the calculator mode based on the adjustments made in the calculator preview mode. The invention further allows the user to embed the calculator in any web page, to email a link to the calculator, and to open and save the calculator.

TABLE 1

```

<html>
  <head>
    <TITLE>make thanksgiving dinner1 - Blox.com BrainMatter
Calculator</TITLE>

    <link rel="stylesheet" type="text/css"
href="/css/static/view_mode_ie_67c1e830.cssx"/><link rel="stylesheet"
type="text/css" href="/css/calculator_ie4_eda7fc0b.cssx"/>

    <style type="text/css">
      /* /home/www/template/static/view_mode_ie.style.tpl */
      #buttonDivision {top: 30; left: 0;}
      #buttonPalette {top: 0; left: 0;height: 30;visibility: visible;}
      #topDivision {top: 0; left: 0;visibility: hidden;}

    </style>

    <script language="JavaScript">

      var kTotalJSIncludes = 13;
      var gNumJSIncludesLoaded = 0;
      var kFromGallery = false;
      var kDomain = "blox.com";
      var kRootUrl = "http://www.blox.com/";

      // hide 'rate this calc' button if we're not opening from
the gallery

      if (!kFromGallery)

```

00377-6894760

```

suppressElements(['btnRate','btnRateText','break5']);

function JSIncludeDoneLoading()
{
    gNumJSIncludesLoaded++;
    includeModalProgressWidget.style.width =
(gNumJSIncludesLoaded/kTotalJSIncludes)*250;
}

// creates stylesheet rules to suppress the specified
elements

function suppressElements(elemIdArray) {
    if (!(document.styleSheets &&
document.styleSheets.length > 0)) return;
    for (var i=0; i < elemIdArray.length; i++)
        document.styleSheets[0].addRule('#' +
elemIdArray[i], 'display:none');
}

</script>

</head>

<body class="body" scroll="no" onLoad="loadDocument()"
onBeforeUnload="if(typeof unloadDocument != 'undefined')unloadDocument()"
onUnload="if(typeof unloadDocument != 'undefined')unloadDocument()"
onResize="resizeDocument()">

<span class='view' id='view' style="width: 100%; height: 100%;">

```

```

        <!-- later write the contents of this span -->
    </span>

```

```

<script language="JavaScript">

```

```

    var content = '<div class="modalLabel">Please wait a moment
(the first time you use BrainMatter this may take a little longer).</div>'
    + '<div class="modalProgressBar">'
    + '<div id="includeModalProgressWidget"
class="modalProgressWidget" style="visibility: visible;">'
    + '</div></div>';

    var progressDialog = '<table id="includeProgressDialog"
width="100%" height="100%">'
    + '<tr valign="middle"><td align="center"><div
class="modal" STYLE="visibility:visible; position:static;">'
    + '<table border="0" width="225" cellpadding="0"
cellpadding="0">'
    + '<tr valign="middle">'
    + '<td bgcolor="#ffcc00"><div
class="modalTopLeftBox"></div></td>'
    + '<td bgcolor="#ffe57d" nowrap><span
class="modalTitle">Loading BrainMatter</span></td>'
    + '</tr>'
    + '<tr>'
    + '<td bgcolor="#ffe57d"><br></td>'
    + '<td><div class="modalContent">' + content + '</div><div
id="modalBottomBar" style="height:10px; overflow:hidden"><BR></div></td>'
    + '</tr>'
    + '</table>'
    + '</div></tr></td></table>';

```



```
//  
  
widths[1] = 399;  
widths[2] = 84;  
widths[3] = 73;  
heights[2] = 19;  
heights[3] = 19;  
heights[4] = 19;  
heights[5] = 19;  
heights[6] = 19;  
heights[7] = 31;  
heights[8] = 19;  
heights[9] = 19;  
heights[10] = 19;  
heights[11] = 19;  
heights[12] = 19;  
heights[13] = 19;  
heights[14] = 19;  
heights[15] = 19;  
heights[16] = 19;  
heights[17] = 19;  
heights[18] = 19;  
heights[19] = 19;  
heights[20] = 31;  
heights[21] = 19;  
heights[22] = 19;  
heights[23] = 19;  
  
}  
  
// after this comment, we add //init cells  
gAppId='003d8cf3_9815';
```

```

gOwnerId='003d76ac_435d';
gAppDescription='make thanksgiving dinner1';
var gCellDataArray=[{entry:'Make Thanksgiving
dinner',viewSize:'12pt',foreColor:'navy',viewFamily:'Verdana',_fontWeight:'bold',
_wrapText:'true',_textAlign:'left',m_row:1,m_col:1,i_nr:'Make Thanksgiving
dinner'},
  {viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_te
xtAlign:'left',m_row:2,m_col:1},
  {entry:'How many total guests are you
having?',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_t
extAlign:'left',m_row:3,m_col:1,i_nr:'How many total guests are you having?'},
  {entry:'12',dynamic:'12',derived:'12',
  i_tb:['e[13][2],e[14][2],e[19][2],e[15][2],e[17][2],e[21][2],e[23][2]]',viewSize:'9
pt',_widthClue:'80',foreColor:'maroon',_textAlign:'right',m_row:3,m_col:2,i_nr:'12
'},
  {viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_te
xtAlign:'left',m_row:4,m_col:1},
  {foreColor:'maroon',viewSize:'9pt',_textAlign:'right',m_row:4,m_col:2},
  {entry:'How many
vegetarians?',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'tru
e',_textAlign:'left',m_row:5,m_col:1,i_nr:'How many vegetarians?'},
  {entry:'2',dynamic:'2',derived:'2',
  i_tb:['e[13][2]]',viewSize:'9pt',_widthClue:'80',foreColor:'maroon',_textAlig
n:'right',m_row:5,m_col:2,i_nr:'2'},
  {viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_te
xtAlign:'left',m_row:6,m_col:1},
  {foreColor:'maroon',viewSize:'9pt',_textAlign:'right',m_row:6,m_col:2},
  {entry:'Traditional Holiday Dinner, or something a little more unusual?
(Traditional, Alternate or
Economy)',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',

```


_textAlign:'left',m_row:7,m_col:1,i_nr:'Traditional Holiday Dinner, or something a little more unusual? (Traditional, Alternate or Economy)'},

{entry:'=menu("Traditional","traditional","Alternate","Alternate","Economy","economy")',dynamic:'__menu("r7c2s0",\ 'Traditional\','traditional\','Alternate\','Alternate\','Economy\','economy\'),'derived:'traditional',

i_tb:'[e[14][1],e[17][1],e[19][1],e[21][1],e[23][1],e[13][2],e[13][3],e[17][3],e[17][2],e[11][1],e[20][1],e[13][1]]',m_cellGUI:'{type:'menu', setting:0, label:'Traditional\'}',locked:'true',viewSize:'9pt',m_dirtyOnInit:'true',foreColor:'maroon',_textAlign:'left',m_row:7,m_col:2,i_nr:'Traditional'},

{viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_textAlign:'left',m_row:8,m_col:1},

{locked:'true',m_row:8,m_col:2},

{entry:'Here\'s what you can whip up in your kitchen:',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_textAlign:'left',m_row:9,m_col:1,i_nr:'Here\'s what you can whip up in your kitchen:'},

{viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_textAlign:'left',m_row:10,m_col:1},

{entry:'=IF(B7 ="Economy","A Basic Holiday Feast",if(B7 ="Alternate","An "& B7 &" Holiday Feast","A "& B7 &" Holiday Feast"))',dynamic:'(_eqr(_cel(\ 'r7c2s0\ '),'\Economy\'))?\ 'A Basic Holiday Feast\':(_eqr(_cel(\ 'r7c2s0\ '),'\Alternate\'))?_concat(_concat(\ 'An \',_cel(\ 'r7c2s0\ ')),\ ' Holiday Feast\')):_concat(_concat(\ 'A \',_cel(\ 'r7c2s0\ ')),\ ' Holiday Feast\'))',derived:'A traditional Holiday Feast',

i_rt:'[e[7][2]]',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontWeight:'bold',_wrapText:'true',_textAlign:'left',m_row:11,m_col:1,i_nr:'A traditional Holiday Feast'},

{_borderBottom:'solid black 1pt',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_textAlign:'left',m_row:12,m_col:1},

```

        {_borderBottom:'solid black
1pt',viewSize:'9pt',_textAlign:'right',m_row:12,m_col:2},
        {textAlign:'left',_borderBottom:'solid black
1pt',viewSize:'9pt',m_row:12,m_col:3},
        {entry:'=IF( B7    ="Traditional","Roast Turkey",IF(    $B$7
        ="Alternate","Cornish Game Hens",IF( $B$7 ="Economy","Chicken
Parmesan","")))',dynamic:'(_eqr(_cel( \r7c2s0\    ),\Traditional\'))?\Roast
Turkey\':(_eqr(_cel( \r7c2s0\    ),\Alternate\'))?\Cornish Game
Hens\':(_eqr(_cel( \r7c2s0\    ),\Economy\'))?\Chicken
Parmesan\':\'))',derived:'Roast Turkey',
        i_tb:'[e[15][1]]',
        i_rt:'[e[7][2]]',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontW
eight:'bold',_wrapText:'true',_textAlign:'left',m_row:13,m_col:1,i_nr:'Roast
Turkey'},
        {entry:'=IF( $B$7 ="Traditional",1.25*(    B3    -    B5    ),IF(
        $B$7 ="Alternate",(    B3    -    B5    ),IF( $B$7
        ="Economy",(    B3    -    B5    )*2,""))',dynamic:'(_eqr(_cel(
\r7c2s0\    ),\Traditional\'))*_mul(1.25,_sub(_cel( \r3c2s0\    ),_cel(
\r5c2s0\    ))):(_eqr(_cel(\r7c2s0\    ),\Alternate\'))*_sub(_cel(
\r3c2s0\    ),_cel( \r5c2s0\    ))):(_eqr(_cel( \r7c2s0\
    ),\Economy\'))*_mul(_sub(_cel( \r3c2s0\    ),_cel( \r5c2s0\
    )),2):\'))',derived:'12.5',
        i_rt:'[e[7][2],e[3][2],e[5][2]]',viewSize:'9pt',_widthClue:'80',_textAlign:'right'
,m_row:13,m_col:2,i_nr:'12.5'},
        {entry:'=IF( $B$7 ="Traditional","pounds",IF(    $B$7
        ="Alternate","hens",IF(    $B$7
        ="Economy","pieces","")))',dynamic:'(_eqr(_cel(    \r7c2s0\
    ),\Traditional\'))?\pounds\':(_eqr(_cel( \r7c2s0\
    ),\Alternate\'))?\hens\':(_eqr(_cel( \r7c2s0\
    ),\Economy\'))?\pieces\':\'))',derived:'pounds',

```

```

i_rt:'[e[7][2]]',textAlign:'left',viewSize:'9pt',m_row:13,m_col:3,i_nr:'pounds'
},
{entry: '=IF( $B$7 ="Traditional","Bread Stuffing",IF( $B$7
="Alternate","Corn Bread",IF( $B$7 ="Economy","Garlic
Breadsticks","")))',dynamic: '(_eqr(_cel( \'r7c2s0\' ),\'Traditional\'))?\'Bread
Stuffing\':(_eqr(_cel(\'r7c2s0\' ),\'Alternate\'))?\'Corn Bread\':(_eqr(_cel(
\'r7c2s0\' ),\'Economy\'))?\'Garlic Breadsticks\':\'')')',derived: 'Bread
Stuffing',
i_tb: '[e[14][2],e[14][3]]',
i_rt: '[e[7][2]]',viewSize: '9pt',foreColor: 'navy',viewFamily: 'Verdana',_fontW
eight: 'bold',_wrapText: 'true',_textAlign: 'left',m_row: 14,m_col: 1,i_nr: 'Bread
Stuffing'},
{entry: '=IF( A14 ="Garlic Breadsticks", B3 *2,if( A14
="", "", B3 ))',dynamic: '(_eqr(_cel( \'r14c1s0\' ),\'Garlic
Breadsticks\'))?_mul(_cel( \'r3c2s0\' ),2):(_eqr(_cel( \'r14c1s0\'
),\''))?\'\':_cel(\'r3c2s0\' ))',derived: '12',
i_rt: '[e[14][1],e[3][2]]',viewFormat: 'number',viewSize: '9pt',_widthClue: '80',
_textAlign: 'right',m_row: 14,m_col: 2,i_nr: '12'},
{entry: '=IF( A14 ="Bread Stuffing","cups",if( A14
="", "", "pieces"))',dynamic: '(_eqr(_cel( \'r14c1s0\' ),\'Bread
Stuffing\'))?\'cups\':(_eqr(_cel( \'r14c1s0\'
),\''))?\'\':\'pieces\'))',derived: 'cups',
i_rt: '[e[14][1]]',textAlign: 'left',viewSize: '9pt',m_row: 14,m_col: 3,i_nr: 'cups'},
{entry: '=if( A13 = "", "", "Soup")',dynamic: '(_eqr(_cel( \'r13c1s0\'
),\''))?\'\':\'Soup\'))',derived: 'Soup',
i_tb: '[e[16][1],e[15][2],e[15][3]]',
i_rt: '[e[13][1]]',viewSize: '9pt',foreColor: 'navy',viewFamily: 'Verdana',_font
Weight: 'bold',_wrapText: 'true',_textAlign: 'left',m_row: 15,m_col: 1,i_nr: 'Soup'},
{entry: '=if( A15 = "", "", B3 )',dynamic: '(_eqr(_cel(
\'r15c1s0\' ),\''))?\'\':_cel(\'r3c2s0\' ))',derived: '12',

```

```

i_rt:[e[15][1],e[3][2]],viewSize:'9pt',_widthClue:'80',_textAlign:'right',m_row:15,m_col:2,i_nr:'12'},
  {entry:'=if( A15 ="" ,"" , "cups" )',dynamic:'(_eqr(_cel( \r15c1s0\ '
),\'))?\'\':\'cups\')',derived:'cups',
  i_rt:[e[15][1]],textAlign:'left',viewSize:'9pt',m_row:15,m_col:3,i_nr:'cups'},
  {entry:'=if( A15 ="" ,"" , " - try a simple corn chowder or spicy bean
soup" )',dynamic:'(_eqr(_cel( \r15c1s0\ '
),\'))?\'\':\' - try a simple corn
chowder or spicy bean soup\')',derived:' - try a simple corn chowder or spicy bean
soup',
  i_rt:[e[15][1]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrapText:'true',_textAlign:'left',m_row:16,m_col:1,i_nr:'- try a simple corn chowder or
spicy bean soup'},
  {entry:'=IF( $B$7 ="Traditional","Side of Vegetables",IF( $B$7
="Alternate","Stuffed Mushrooms",IF( $B$7 ="Economy","Side of
Vegetables","")))',dynamic:'(_eqr(_cel( \r7c2s0\ '
),\'Traditional\'))?\\'Side of
Vegetables\':(_eqr(_cel( \r7c2s0\ '
),\'Alternate\'))?\\'Stuffed
Mushrooms\':(_eqr(_cel( \r7c2s0\ '
),\'Economy\'))?\\'Side of
Vegetables\':\'))',derived:'Side of Vegetables',
  i_tb:[e[17][3],e[18][1],e[17][2]],
  i_rt:[e[7][2]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontWeight:'bold',_wrapText:'true',_textAlign:'left',m_row:17,m_col:1,i_nr:'Side of
Vegetables'},
  {entry:'=IF( B7 ="Alternate",3* B3 ,if( A17 ="" ,"" ,1*
B3 ))',dynamic:'(_eqr(_cel( \r7c2s0\ '
),\'Alternate\'))?_mul(3,_cel(
\r3c2s0\ '
)):(_eqr(_cel( \r17c1s0\ '
),\'))?\'\':_mul(1,_cel(
\r3c2s0\ '
)))',derived:'12',
  i_rt:[e[7][2],e[3][2],e[17][1]],viewFormat:'number',viewSize:'9pt',_widthClue:'80',_textAlign:'right',m_row:17,m_col:2,i_nr:'12'},
  {entry:'=IF( B7 ="Alternate","mushrooms",if( A17
=""" ,"" , "cups" )',dynamic:'(_eqr(_cel( \r7c2s0\ '

```

```

),\Alternate\')?\mushrooms\':(_eqr(_cel( \r17c1s0\
),\')?\'\':\cups\'))',derived:'cups',
i_rt:[e[7][2],e[17][1]],textAlign:'left',viewSize:'9pt',m_row:17,m_col:3,i_nr:'
cups'},
{entry:'=if( A17 ="" ,"" , " - try green beans, squash, or a vegetable
medley as well")',dynamic:'(_eqr(_cel( \r17c1s0\ ),\')?\'\':\ - try green
beans, squash, or a vegetable medley as well\')',derived:' - try green beans, squash,
or a vegetable medley as well',
i_rt:[e[17][1]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrap
Text:true',_textAlign:'left',m_row:18,m_col:1,i_nr:'- try green beans, squash, or a
vegetable medley as well'},
{entry:'=IF( $B$7 ="Traditional","Side of Potatoes",IF( $B$7
="Alternate","Yams",IF( $B$7
="Economy","Yams","")))',dynamic:'(_eqr(_cel( \r7c2s0\
),\Traditional\')?\Side of Potatoes\':(_eqr(_cel( \r7c2s0\
),\Alternate\')?\Yams\':(_eqr(_cel( \r7c2s0\
),\Economy\')?\Yams\':\'))',derived:'Side of Potatoes',
i_tb:[e[19][2],e[19][3],e[20][1]],
i_rt:[e[7][2]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontW
eight:'bold',_wrapText:true',_textAlign:'left',m_row:19,m_col:1,i_nr:'Side of
Potatoes'},
{entry:'=if( A19 ="" ,"" , B3 /2)',dynamic:'(_eqr(_cel(
\r19c1s0\ ),\')?\'\':_div(_cel( \r3c2s0\ ),2))',derived:'6',
i_rt:[e[19][1],e[3][2]],viewFormat:'number',viewSize:'9pt',_widthClue:'80',
_textAlign:'right',m_row:19,m_col:2,i_nr:'6'},
{entry:'=if( A19 ="" ,"" , "pounds")',dynamic:'(_eqr(_cel( \r19c1s0\
),\')?\'\':\pounds\')',derived:'pounds',
i_rt:[e[19][1]],textAlign:'left',viewSize:'9pt',m_row:19,m_col:3,i_nr:'pound
s'},
{entry:'=IF( B7 ="Traditional"," - there\'s more than one way to cook a
potato! Try scalloped, mashed, twice-baked, or even au gratin!",if( A19

```

=""", """, " - yams are sweeter than potatoes and can be prepared just as easily,
for a different twist on the holidays!"),dynamic:'(_eqr(_cel(\r7c2s0\'
),\Traditional\'))?\' - there\\\'s more than one way to cook a potato! Try
scalloped, mashed, twice-baked, or even au gratin!\':(_eqr(_cel(\r19c1s0\'
),\'))?\'\'\' - yams are sweeter than potatoes and can be prepared just as
easily, for a different twist on the holidays!\'))',derived:' - there\'s more than one
way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!'},
i_rt:[e7][2],e[19][1]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana'
,_wrapText:'true',_textAlign:'left',m_row:20,m_col:1,i_nr:'- there\'s more than one
way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!'},
{entry:'=IF(\$B\$7 ="Traditional","Pumpkin Pie",IF(\$B\$7
="Alternate","Pecan Pie",IF(\$B\$7 ="Economy","Pumpkin
Pie","")))',dynamic:'(_eqr(_cel(\r7c2s0\'),\Traditional\'))?\'Pumpkin
Pie\':(_eqr(_cel(\r7c2s0\'),\Alternate\'))?\'Pecan Pie\':(_eqr(_cel(
\r7c2s0\'),\Economy\'))?\'Pumpkin Pie\':\'))',derived:'Pumpkin Pie',
i_tb:[e21][2],e[21][3],e[22][1]],
i_rt:[e7][2]],viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontW
eight:'bold',_wrapText:'true',_textAlign:'left',m_row:21,m_col:1,i_nr:'Pumpkin
Pie'},
{entry:'=if(A21 =""""",ROUND(B3 /8,0))',dynamic:'(_eqr(_cel(
\r21c1s0\'),\'))?\'\'\'__round(_div(_cel(\r3c2s0\'
),8),0))',derived:'2',
i_rt:[e21][1],e[3][2]],viewFormat:'number',viewSize:'9pt',_widthClue:'80',
_textAlign:'right',m_row:21,m_col:2,i_nr:'2'},
{entry:'=if(A21 =""""", """, "pies")',dynamic:'(_eqr(_cel(\r21c1s0\'
),\'))?\'\'\'\'pies\')',derived:'pies',
i_rt:[e21][1]],textAlign:'left',viewSize:'9pt',m_row:21,m_col:3,i_nr:'pies'},
{entry:'=if(A21 =""""", """, " - or try something different, like a Sweet
Potato Pie!')',dynamic:'(_eqr(_cel(\r21c1s0\'),\'))?\'\'\' - or try something
different, like a Sweet Potato Pie!\')',derived:' - or try something different, like a
Sweet Potato Pie!'},

i_rt:'[e[21][1]]',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_wrap
Text:true',_textAlign:'left',m_row:22,m_col:1,i_nr:'- or try something different,
like a Sweet Potato Pie!'},

{entry:'=IF(\$B\$7 ="Traditional","Hot Apple Cider",IF(\$B\$7
="Alternate","Spiced Cranberry Cider",IF(\$B\$7 ="Economy","Hot
Apple Cider","")))',dynamic:'(_eqr(_cel(\'r7c2s0\'),\'Traditional\'))?\'Hot
Apple Cider\':(_eqr(_cel(\'r7c2s0\'),\'Alternate\'))?\'Spiced Cranberry
Cider\':(_eqr(_cel(\'r7c2s0\'),\'Economy\'))?\'Hot Apple
Cider\':\''))',derived:'Hot Apple Cider',

i_tb:'[e[23][2],e[23][3]]',

i_rt:'[e[7][2]]',_borderBottom:'solid black
1px',viewSize:'9pt',foreColor:'navy',viewFamily:'Verdana',_fontWeight:'bold',_wra
pText:true',_textAlign:'left',m_row:23,m_col:1,i_nr:'Hot Apple Cider'},

{entry:'=if(A23 =""",", B3 *2)',dynamic:'(_eqr(_cel(
'r23c1s0\'),\''))?\'\'':_mul(_cel(\'r3c2s0\'),2))',derived:'24',

i_rt:'[e[23][1],e[3][2]]',_borderBottom:'solid black

1px',viewFormat:'number',viewSize:'9pt',_widthClue:'80',_textAlign:'right',m_row
:23,m_col:2,i_nr:'24'},

{entry:'=if(A23 =""",", "cups")',dynamic:'(_eqr(_cel(\'r23c1s0\'
,\''))?\'\'':\'cups\')',derived:'cups',

i_rt:'[e[23][1]]',textAlign:'left',_borderBottom:'solid black

1px',viewSize:'9pt',m_row:23,m_col:3,i_nr:'cups'}];

var needsRebuilding=false;

var gDerivedFrom="";

var kSaveHostname = "";

gMacroText= "";


```

<div id="btnEmail" class="button" title="Email to others"></div><div
id="btnEmailText" class="buttonText">Email to others</div>
<div id="break5" class="break"></div>
<div id="btnRate" class="button" title="Rate this calculator"></div><div
id="btnRateText" class="buttonText">Rate this calculator</div>
</div>
<div id="buttonDivision" class="division"></div>
<div id="footDivision" class="division"></div>
<div id="footPalette" class="palette">
<nobr>Copyright &copy; 1999-2000 <a href="http://www.alphablox.com"
target="_blank">AlphaBlox Corporation Inc.</a> &trade;</nobr></div>

<div id="modalDialog" style="display:none"></div>
<div id="popupMenuList" class="smallList" style="width:100;
height:1;"></div>

<script language="JavaScript">
    includeProgressDialog.style.display = "none";
</script>

<iframe id=resource0 src="" style="display: none;"></iframe>
<iframe id=resource1 src="" style="display: none;"></iframe>
<iframe id=resource2 src="" style="display: none;"></iframe>
<iframe id=resource3 src="" style="display: none;"></iframe>

<iframe id="saveFrame" src="/save_frame.html" style="display:
none;"></iframe>
<iframe id="macroFrame" src="/macro_frame" style="display:
none;"></iframe>
<iframe id="bugFrame" src="" style="display: none;"></iframe>

```

[illegible]

the ultimate holiday dinner - Blox.com BrainMatter Spreadsheet				
File Edit Tools Format Help				
B13: f(x) * [Icons] [9pt] [Arial] [color] [email this page]				
1	Make the Ultimate Holiday Dinner -- 104			
2				
3	How many total guests are you having?	12-106		
4				
5	How many vegetarians?	2-108		
6				
7	Traditional Holiday Dinner, or something a little more unusual? (Traditional, Alternate or Economy)	Traditional * -110		
8				
9	Here's what you can whip up in your kitchen:			
10				
11	A traditional Holiday Feast	-111		
12				
13	Roast Turkey	12.5 pounds		
14	Bread Stuffing	12 cups		
15	Soup	12 cups		
16	- try a simple corn chowder or spicy bean soup			
17	Side of Vegetables	12 cups		
18	- try green beans, squash, or a vegetable medley as well			
19	Side of Potatoes	6 pounds		
20	- there's more than one way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!			

FIG. 1

Make the Ultimate Holiday Dinner

How many total guests are you having?

12 - 106

How many vegetarians?

2 - 108

Traditional Holiday Dinner, or something a little more unusual?
(Traditional, Alternate or Economy)

Traditional - 110

Here's what you can whip up in your kitchen:

A traditional Holiday Feast

Roast Turkey	304	12.5 pounds	111
Bread Stuffing		12 cups	
Soup		12 cups	
- try a simple corn chowder or spicy bean soup			
Side of Vegetables		12 cups	
- try green beans, squash, or a vegetable medley as well			
Side of Potatoes		6 pounds	
- there's more than one way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!			
Pumpkin Pie		2 pies	
- or try something different, like a Sweet Potato Pie!			
Hot Apple Cider		24 cups	

Make the Ultimate Holiday Dinner

How many total guests are you having? **106**

How many vegetarians? **108**

Traditional Holiday Dinner, or something a little more unusual? **302** **110**

Here's what you can whip up in your kitchen:

A traditional Holiday Feast

Roast Turkey	12.5 pounds
Bread Stuffing	12 cups
Soup	12 cups
- try a simple corn chowder or spicy bean soup	
Side of Vegetables	12 cups
- try green beans, squash, or a vegetable medley as well	
Side of Potatoes	6 pounds
- there's more than one way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!	
Pumpkin Pie	2 pies
- or try something different, like a Sweet Potato Pie!	
Hot Apple Cider	24 cups

Fig 3

410

the ultimate holiday dinner - Blox.com BrainMatter Calculator - Microsoft Internet Explorer

Make the Ultimate Holiday Dinner - 402

How many total guests are you having?

12 - 106

How many vegetarians?

2 - 108

Traditional Holiday Dinner, or something a little more unusual?
(Traditional, Alternate or Economy)

Traditional - 110

Here's what you can whip up in your kitchen:

A traditional Holiday Feast

Roast Turkey

12.5 pounds

Bread Stuffing

12 cups

Soup

12 cups

- try a simple corn chowder or spicy bean soup

Side of Vegetables

12 cups

- try green beans, squash, or a vegetable medley as well

Side of Potatoes

6 pounds

- there's more than one way to cook a potato! Try scalloped, mashed, twice-baked, or even au gratin!

Pumpkin Pie

2 pies

- or try something different, like a Sweet Potato Pie!

Hot Apple Cider

24 cups

408

406

400

Fig. 4

[illegible]

—

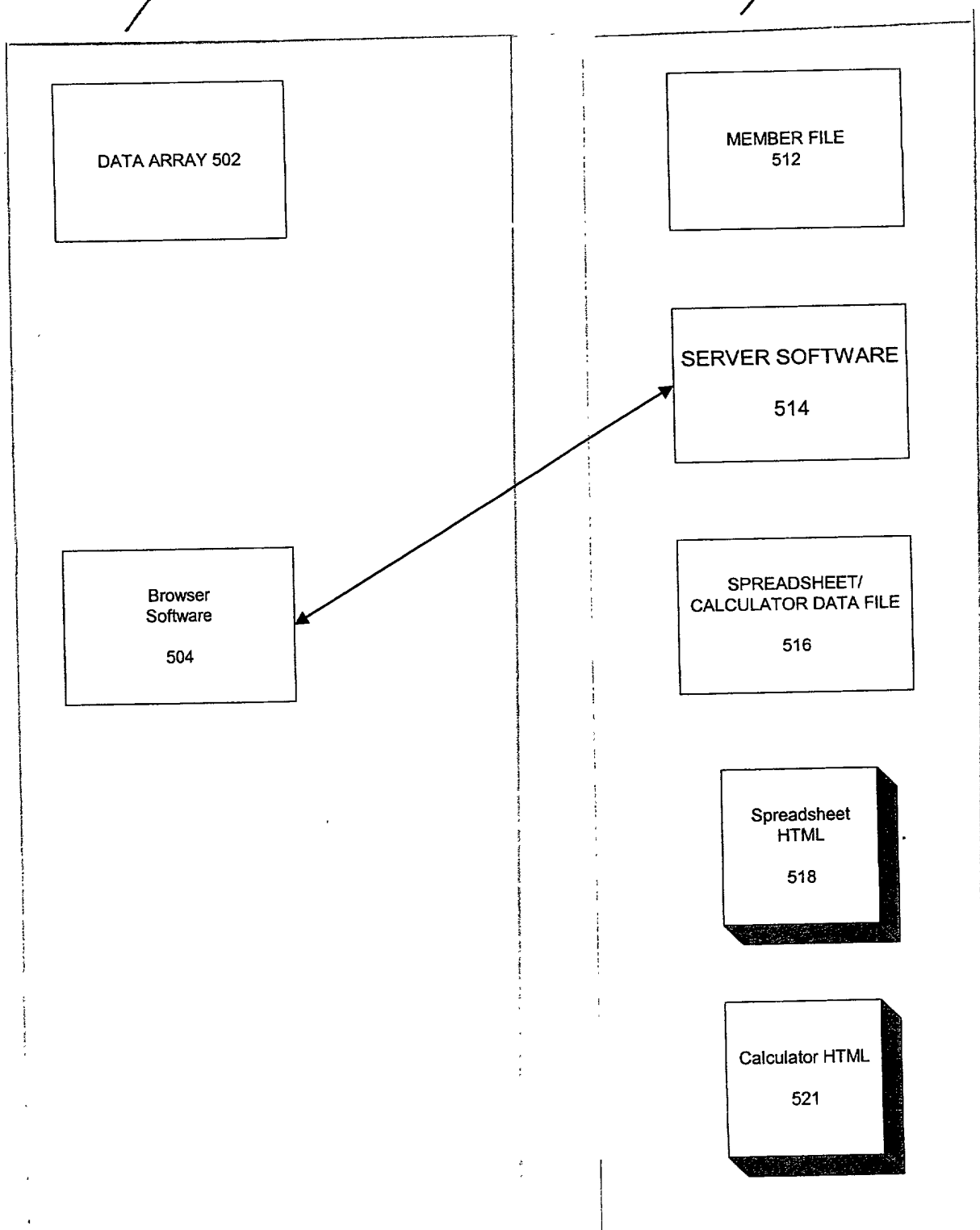


FIG. 5

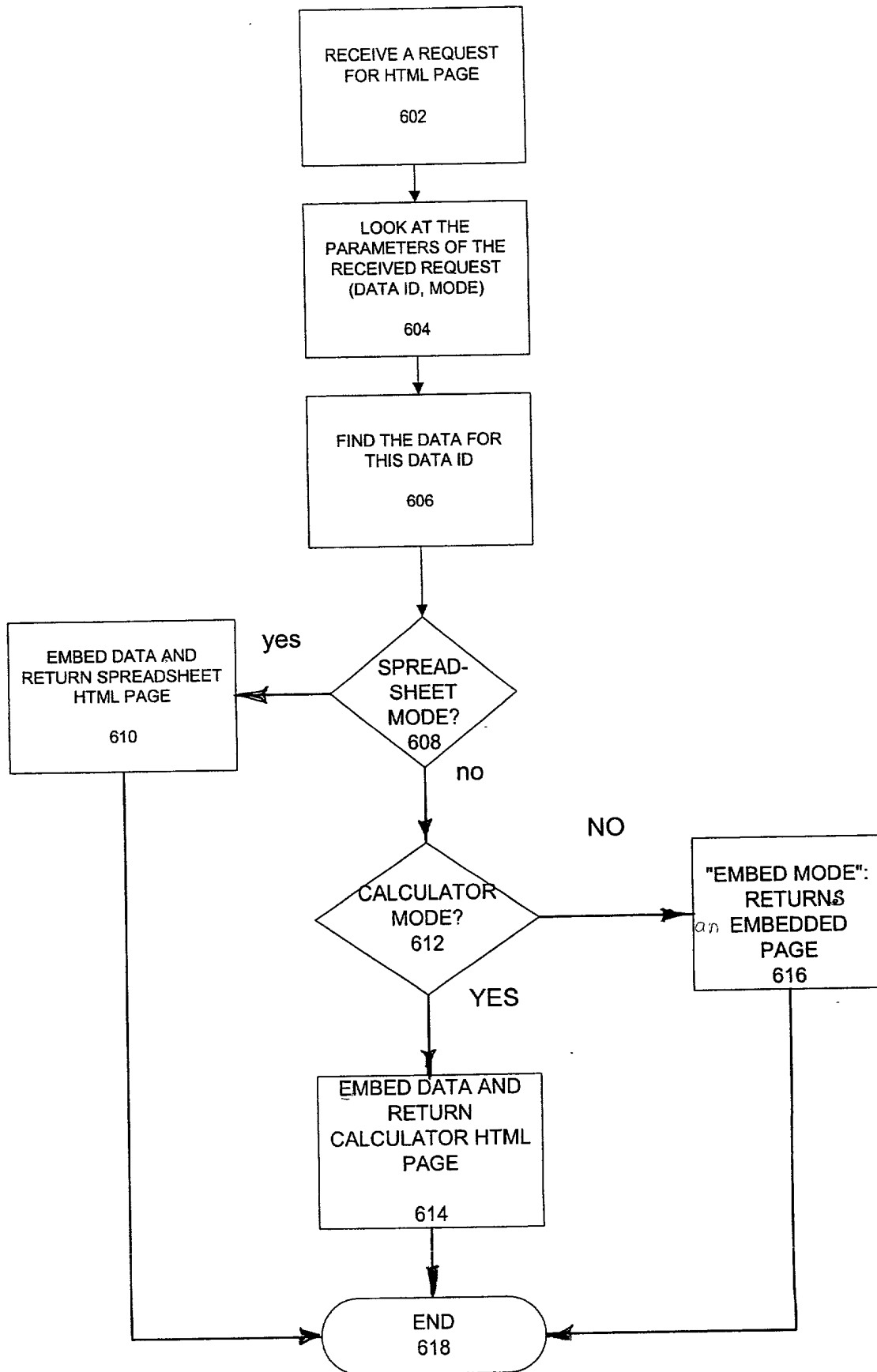


FIG. 6A

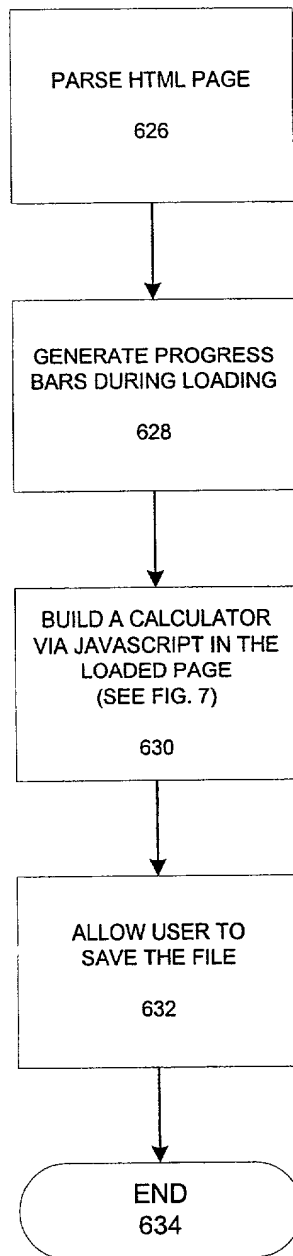


FIG. 6B

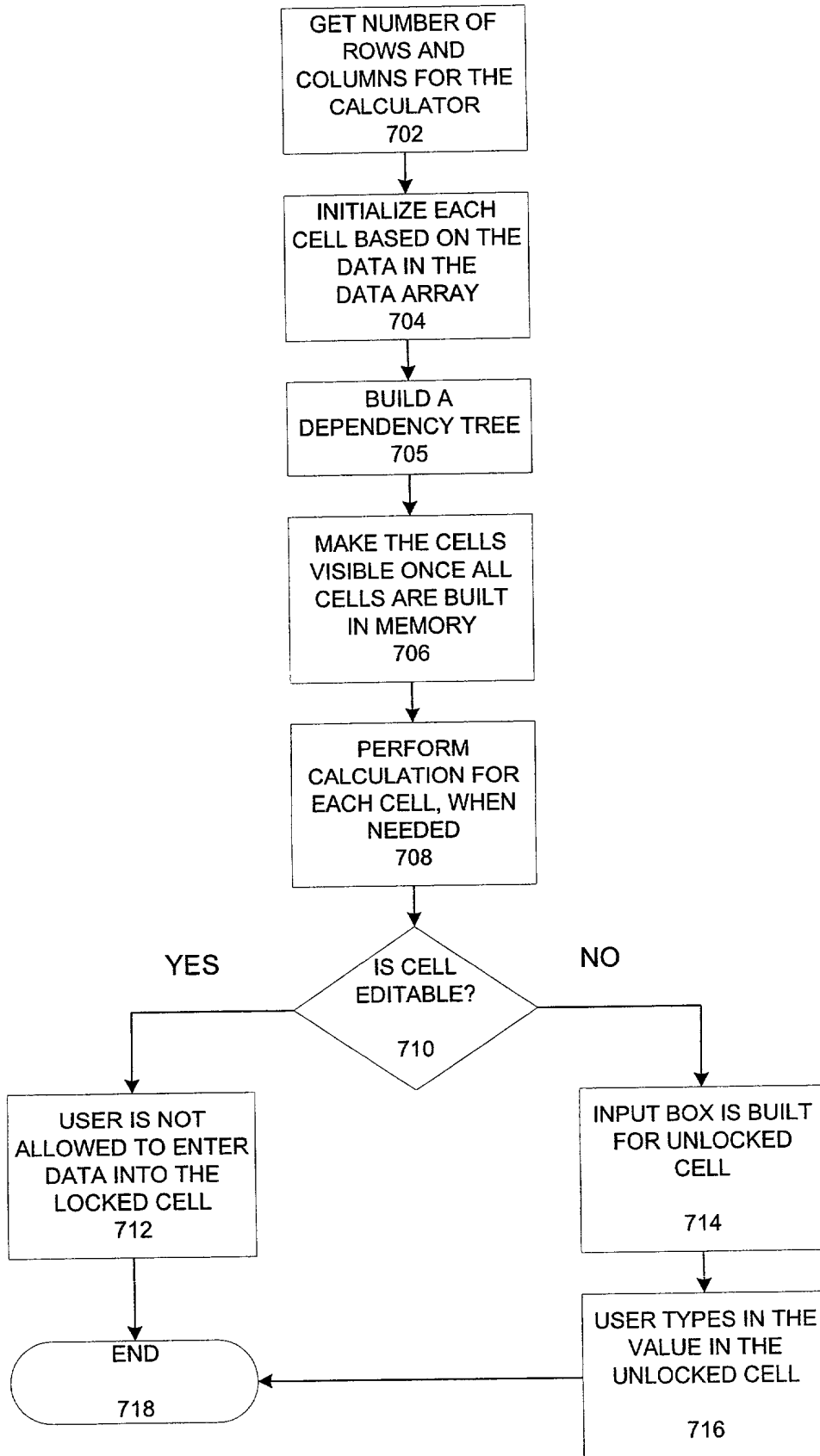
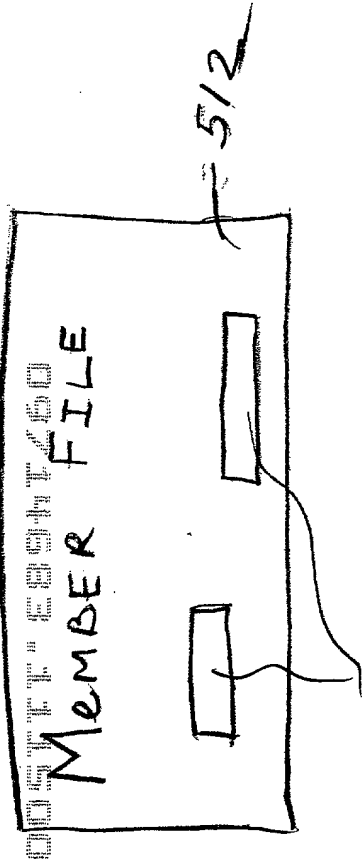
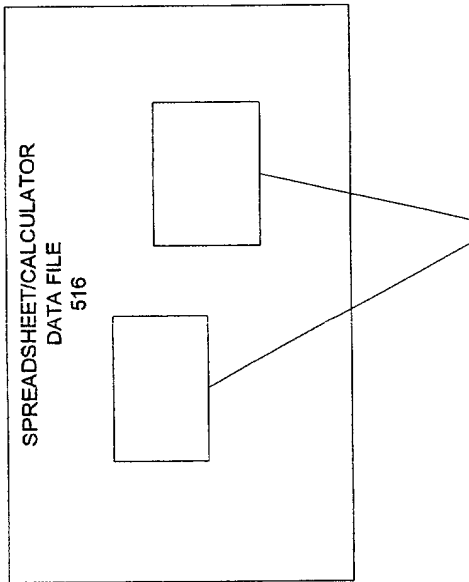


FIG. 7



MEMBER ID	password	e-mail address	data ID
Anna	0124	Anna@hotmail.com	Make the Ultimate Holiday Dinner

Fig 8



900

INFORMATION ABOUT EACH CELL	
USER NAME	DATA ID
ANNA	HOW MANY TOTAL GUESTS ARE YOU HAVING? CELL A3: {ENTRY:'HOW MANY TOTAL GUESTS ARE YOU HAVING?',LOCKED:FALSE,VIEWSIZE:9PT,FORECOLOR:NAVY,VIEWFAMILY:VERDANA,_WRAPTEXT:TRUE,_TEXTALIGN:LEFT,M_ROW:3,M_COL:1,I_NR:'HOW MANY TOTAL GUESTS ARE YOU HAVING?'},
	HOW MANY VEGETARIANS? CELL A5: {ENTRY:'HOW MANY VEGETARIANS?',LOCKED:FALSE,VIEWSIZE:9PT,FORECOLOR:NAVY,VIEWFAMILY:VERDANA,_WRAPTEXT:TRUE,_TEXTALIGN:LEFT,M_ROW:5,M_COL:1,I_NR:'HOW MANY VEGETARIANS?'},

FIG. 9

00511633 11300

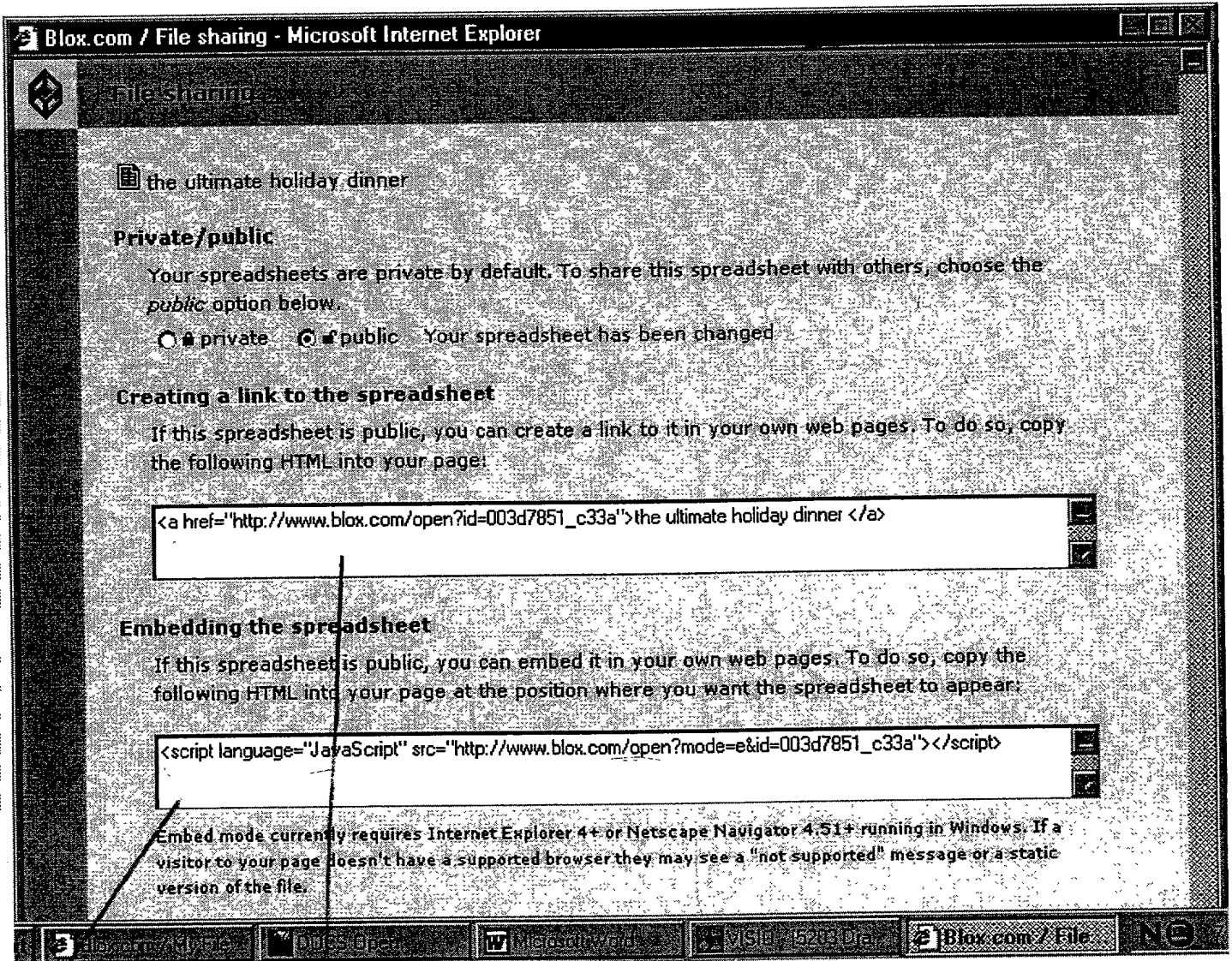


Fig. 10

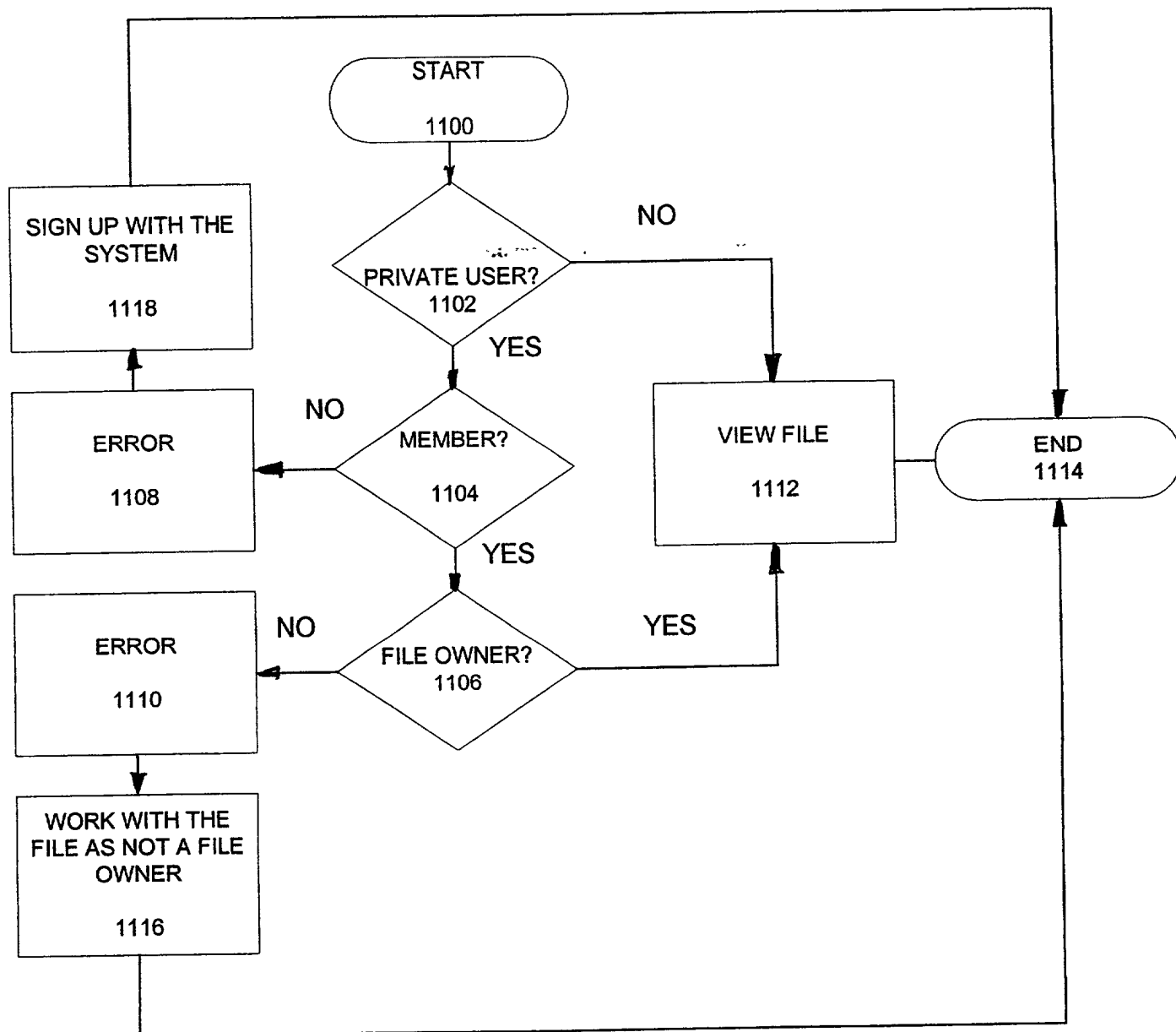


FIG. 11

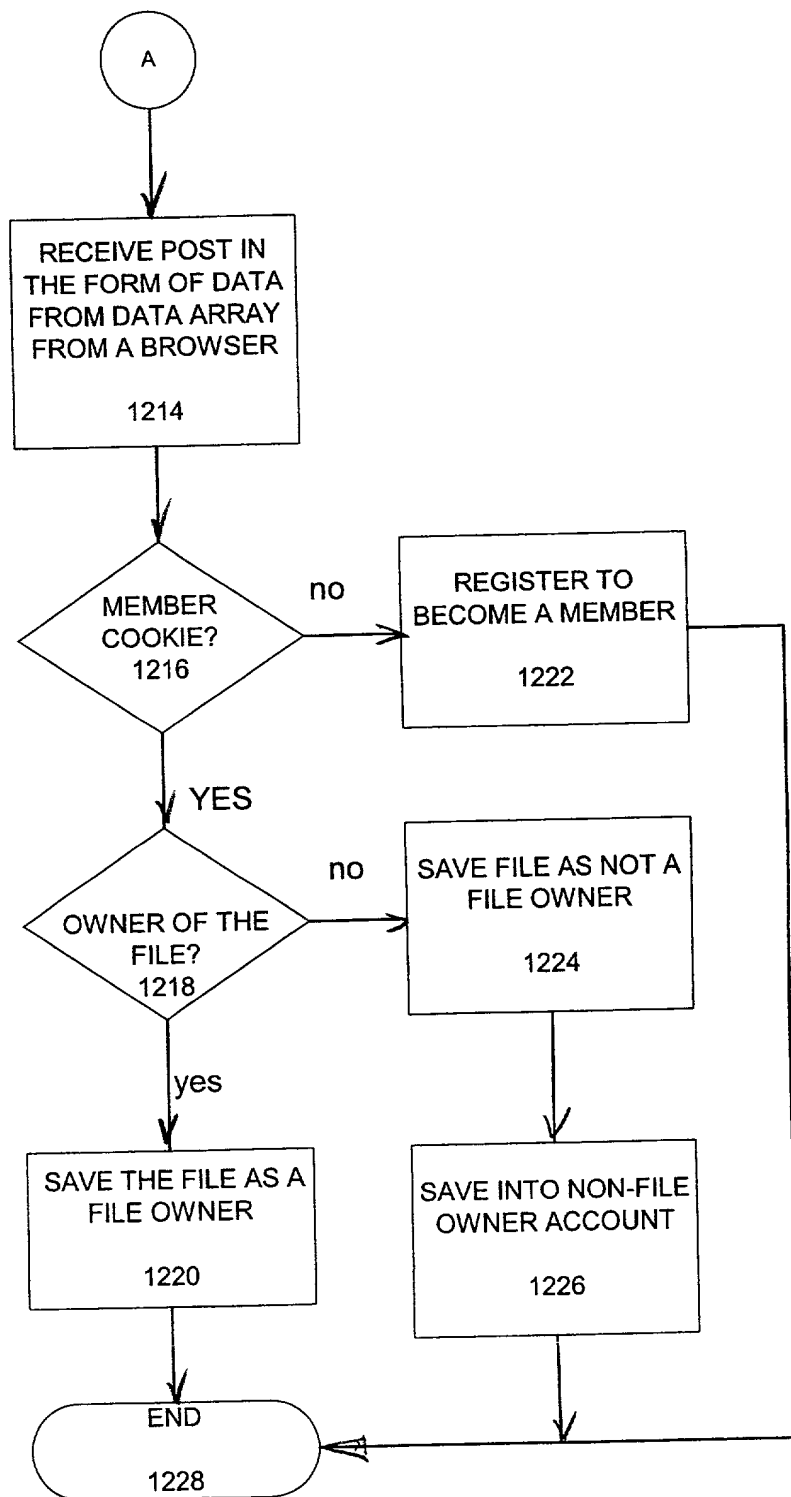
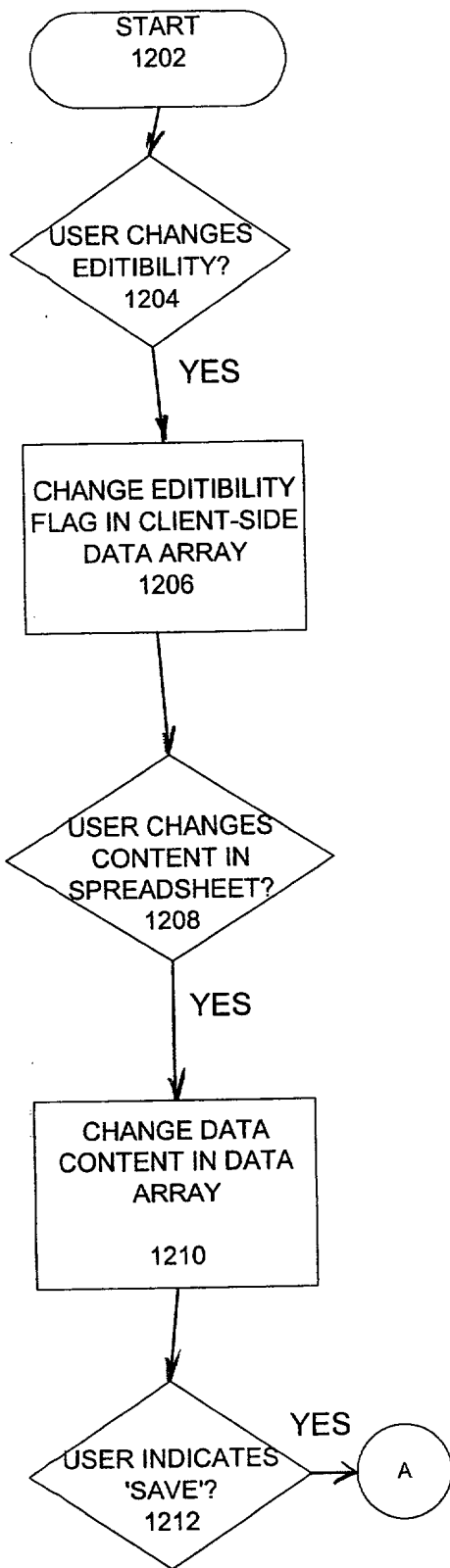


FIG. 12

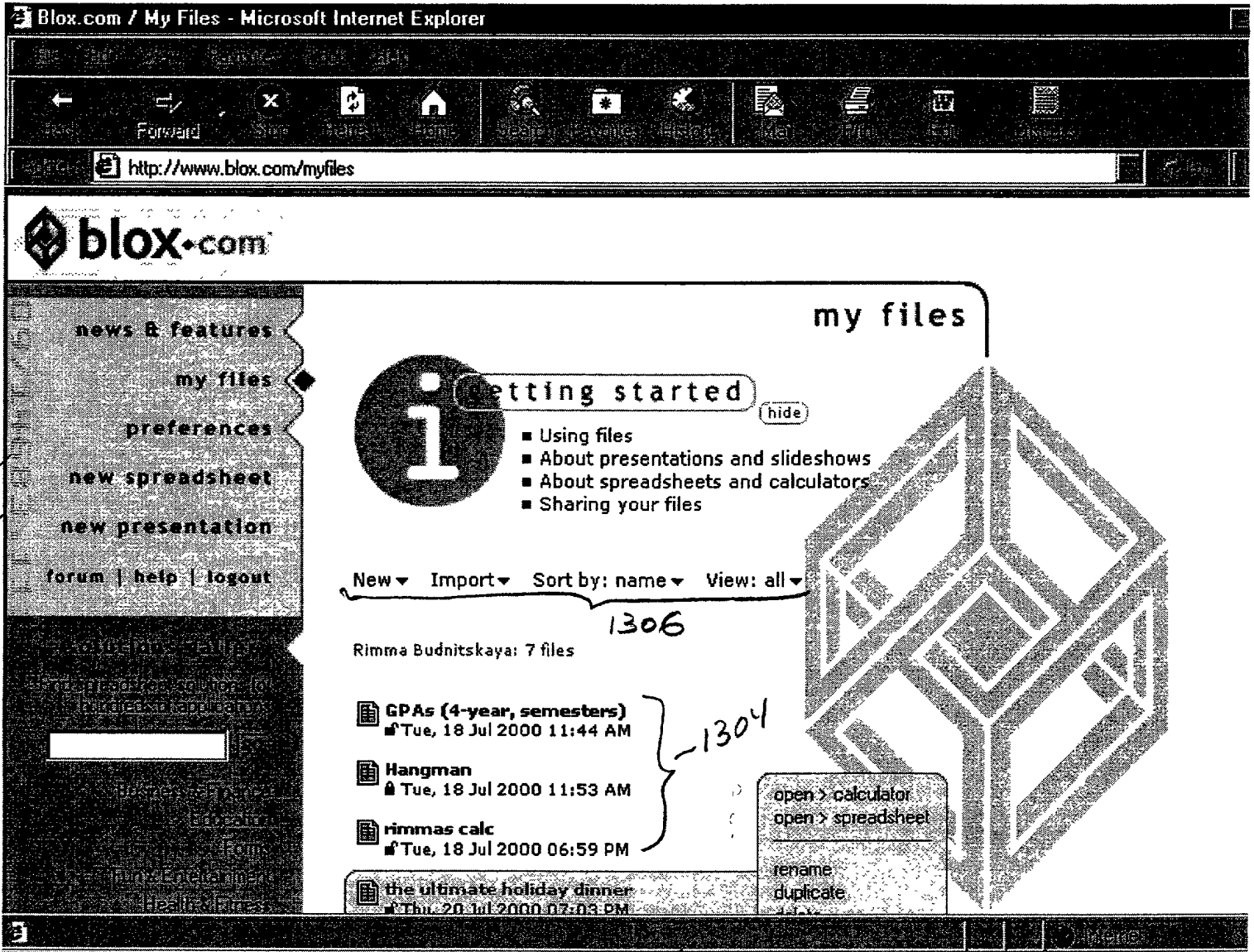


Fig 13

0010/PTO Rev. 6/95 U.S. Department of Commerce Patent and Trademark Office DECLARATION FOR UTILITY OR DESIGN PATENT APPLICATION <input checked="" type="checkbox"/> Declaration Submitted with Initial Filing OR <input type="checkbox"/> Declaration Submitted after Initial Filing	Attorney Docket Number	5481 US
	First Named Inventor	Guttman
	<i>COMPLETE IF KNOWN</i>	
	Application Number	To Be Assigned
	Filing Date	Herewith
	Group Art Unit	To Be Assigned
	Examiner Name	To Be Assigned

As a below named inventor, I hereby declare that:
My residence, post office address, and citizenship are as stated below next to my name.
I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

CLIENT SIDE, WEB-BASED CALCULATOR

the specification of which *(Title of the Invention)*
☒ is attached hereto
OR
☐ was filed on (MM/DD/YYYY) [] as United States Application Number or PCT International Application Number [] and was amended on (MM/DD/YYYY) [] (if applicable).
I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment specifically referred to above.
I acknowledge the duty to disclose information which is material to patentability as defined in Title 37 Code of Federal Regulations. § 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code § 119 (a)-(d) or § 365(b) of any foreign application(s) for patent or inventor's certificate, or § 365 (a) of any PCT international application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or of any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application Number(s)	Country	Foreign Filing Date (MM/DD/YYYY)	Priority Not Claimed	Certified Copy Attached?	
				YES	NO
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

☐ Additional foreign application numbers are listed on a supplemental priority sheet attached hereto:

I hereby claim the benefit under Title 35, United States Code § 119(e) of any United States provisional application(s) listed below.

Application Number(s)	Filing Date (MM/DD/YYYY)	<input type="checkbox"/> Additional provisional application numbers are listed on a supplemental sheet attached hereto.
60/241,083	10/16/2000	

00514347260

DECLARATION			Page 2
I hereby claim the benefit under Title 35, United States Code § 120 of any United States application(s), or § 365(c) of any PCT international application designating the United States of America, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT international application in the manner provided by the first paragraph of Title 35, United States Code § 112, I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations § 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application.			
U.S. Parent Application Number	PCT Parent Number	Parent Filing Date (MM/DD/YYYY)	Parent Patent Number (if applicable)
<input type="checkbox"/> Additional U.S. or PCT international application numbers are listed on a supplemental priority sheet attached hereto.			

As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith:			
Name	Registration Number	Name	Registration Number
Greg T. Sueoka	33,800		
Laura A. Majerus	33,417		
Charles E. Schulman	43,350		
Deepti Panchawagh-Jain	43,846		
<input type="checkbox"/> Additional attorney(s) and/or agent(s) named on a supplemental sheet attached hereto.			
Please direct all correspondence to:			
<p style="text-align: center;"> Laura A. Majerus Fenwick & West LLP Two Palo Alto Square Palo Alto, CA 94306 U.S.A. </p>			
Telephone	(650) 858-7152	Fax	(650) 494-1417

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.							
Name of Sole or First Inventor:		<input type="checkbox"/> A petition has been filed for this unsigned inventor					
Given Name	Steve	Middle Initial		Family Name	Guttman	Suffix e.g. Jr.	
Inventor's Signature					Date		
Residence. City	San Francisco	State	CA	Country	USA	Citizenship	US
Mailing Address							
City	San Francisco	State	CA	Zip	94111	Country	USA
<input type="checkbox"/> Additional inventors are being named on supplemental sheet(s) attached hereto							

DECLARATION					ADDITIONAL INVENTOR(S) Supplemental Sheet			
Name of Additional Joint Inventor, if any:					<input type="checkbox"/> A petition has been filed for this unsigned inventor			
Given Name	Joseph		Middle Initial		Family Name	Ternasky		Suffix e.g. Jr.
Inventor's Signature						Date		
Residence: City	Mountain View		State	CA	Country	USA		Citizenship US
Mailing Address	450 Sierra Vista Avenue, Apt. #6							
Mailing Address								
City	Mountain View		State	CA	Zip	94043		Country USA

Name of Additional Joint Inventor, if any:					<input type="checkbox"/> A petition has been filed for this unsigned inventor			
Given Name			Middle Initial		Family Name			Suffix e.g. Jr.
Inventor's Signature						Date		
Residence: City			State		Country			Citizenship
Mailing Address								
Mailing Address								
City			State		Zip			Country

Name of Additional Joint Inventor, if any:					<input type="checkbox"/> A petition has been filed for this unsigned inventor			
Given Name			Middle Initial		Family Name			Suffix e.g. Jr.
Inventor's Signature						Date		
Residence: City			State		Country			Citizenship
Mailing Address								
Mailing Address								
City			State		Zip			Country

Name of Additional Joint Inventor, if any:					<input type="checkbox"/> A petition has been filed for this unsigned inventor			
Given Name			Middle Initial		Family Name			Suffix e.g. Jr.
Inventor's Signature						Date		
Residence: City			State		Country			Citizenship
Mailing Address								
Mailing Address								
City			State		Zip			Country
<input type="checkbox"/> Additional inventors are being named on supplemental sheet(s) attached hereto								